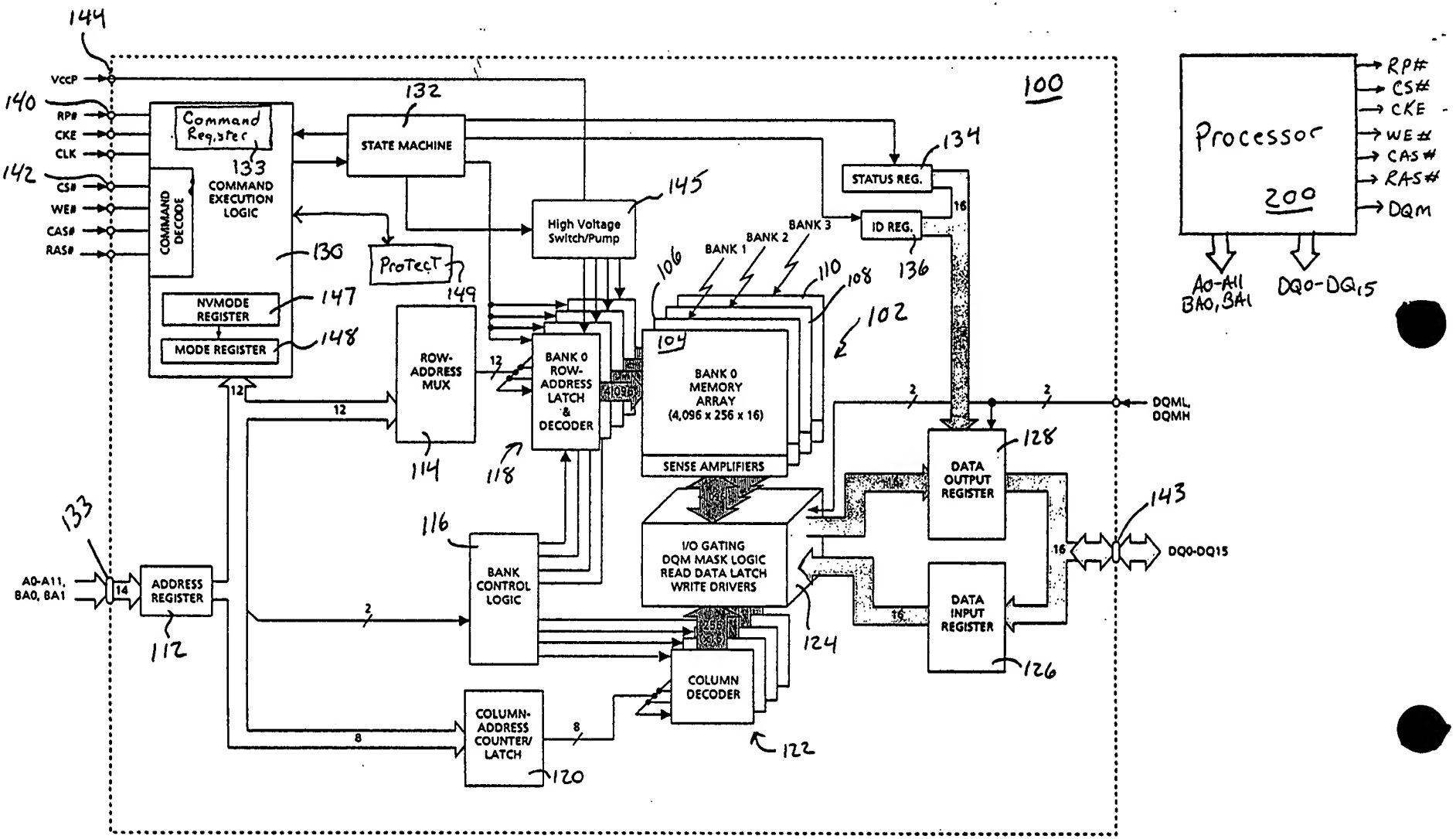

144
VCCP
140
RP#
CKE
CLK
142
CS#
WE#
CAS#
RAS#
COMMAND DECODE
Command Register
133
COMMAND EXECUTION LOGIC
132
STATE MACHINE
130
Protect
149
NVMODE REGISTER
147
MODE REGISTER
148
145
High Voltage Switch/Pump
134
STATUS REG.
ID REG.
136
16
100
BANK 1
BANK 2
BANK 3
106
110
108
102
104
ROW-ADDRESS MUX
12
BANK 0 ROW-ADDRESS LATCH & DECODER
BANK 0 MEMORY ARRAY (4,096 x 256 x 16)
SENSE AMPLIFIERS
114
118
116
BANK CONTROL LOGIC
I/O GATING DQM MASK LOGIC READ DATA LATCH WRITE DRIVERS
DATA OUTPUT REGISTER
128
DATA INPUT REGISTER
126
124
DQML
DQMH
143
DQ0-DQ15
133
A0-A11, BA0, BA1
ADDRESS REGISTER
112
COLUMN DECODER
122
COLUMN-ADDRESS COUNTER/ LATCH
120
Processor
200
RP#
CS#
CKE
WE#
CAS#
RAS#
DQM
A0-A11 BA0, BA1
DQ0-DQ15

Fig. 1A

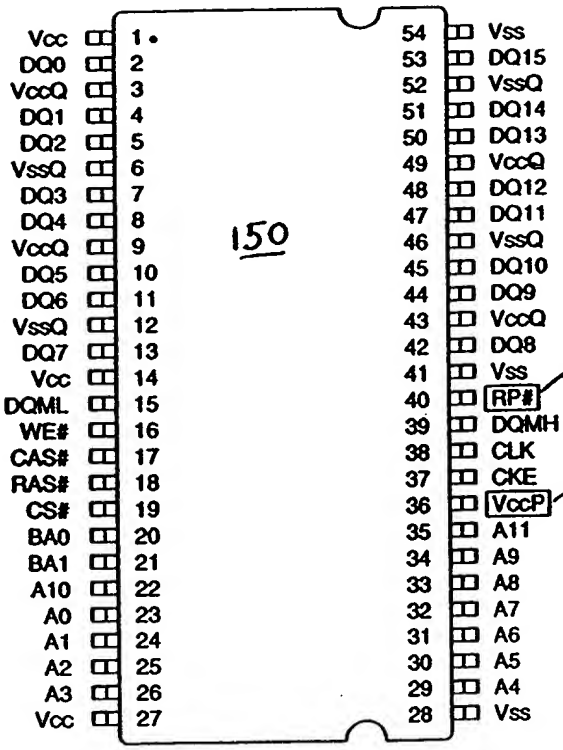

Vcc 1
DQ0 2
VccQ 3
DQ1 4
DQ2 5
VssQ 6
DQ3 7
DQ4 8
VccQ 9
DQ5 10
DQ6 11
VssQ 12
DQ7 13
Vcc 14
DQML 15
WE# 16
CAS# 17
RAS# 18
CS# 19
BA0 20
BA1 21
A10 22
A0 23
A1 24
A2 25
A3 26
Vcc 27
54 Vss
53 DQ15
52 VssQ
51 DQ14
50 DQ13
49 VccQ
48 DQ12
47 DQ11
46 VssQ
45 DQ10
44 DQ9
43 VccQ
42 DQ8
41 Vss
40 RP#
39 DQMH
38 CLK
37 CKE
36 VccP
35 A11
34 A9
33 A8
32 A7
31 A6
30 A5
29 A4
28 Vss
150
152
154

Fig. 1B

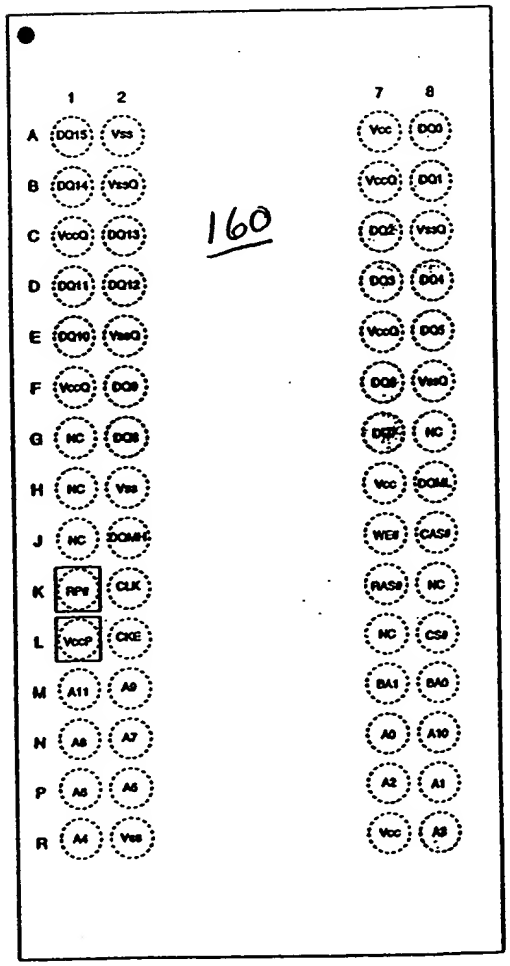

1 2
7 8
A DQ15 Vss Vcc DQ0
B DQ14 VssQ VccQ DQ1
C VccQ DQ13 DQ2 VssQ
D DQ11 DQ12 DQ3 DQ4
E DQ10 VssQ VccQ DQ5
F VccQ DQ9 DQ6 VssQ
G NC DQ8 DQ7 NC
H NC Vss Vcc DQML
J NC DQMH WE# CAS#
K RP# CLK RAS# NC
L VccP CKE NC CS#
M A11 A9 BA1 BA0
N A8 A7 A0 A10
P A6 A5 A2 A1
R A4 Vss Vcc A3
160

Fig. 1C

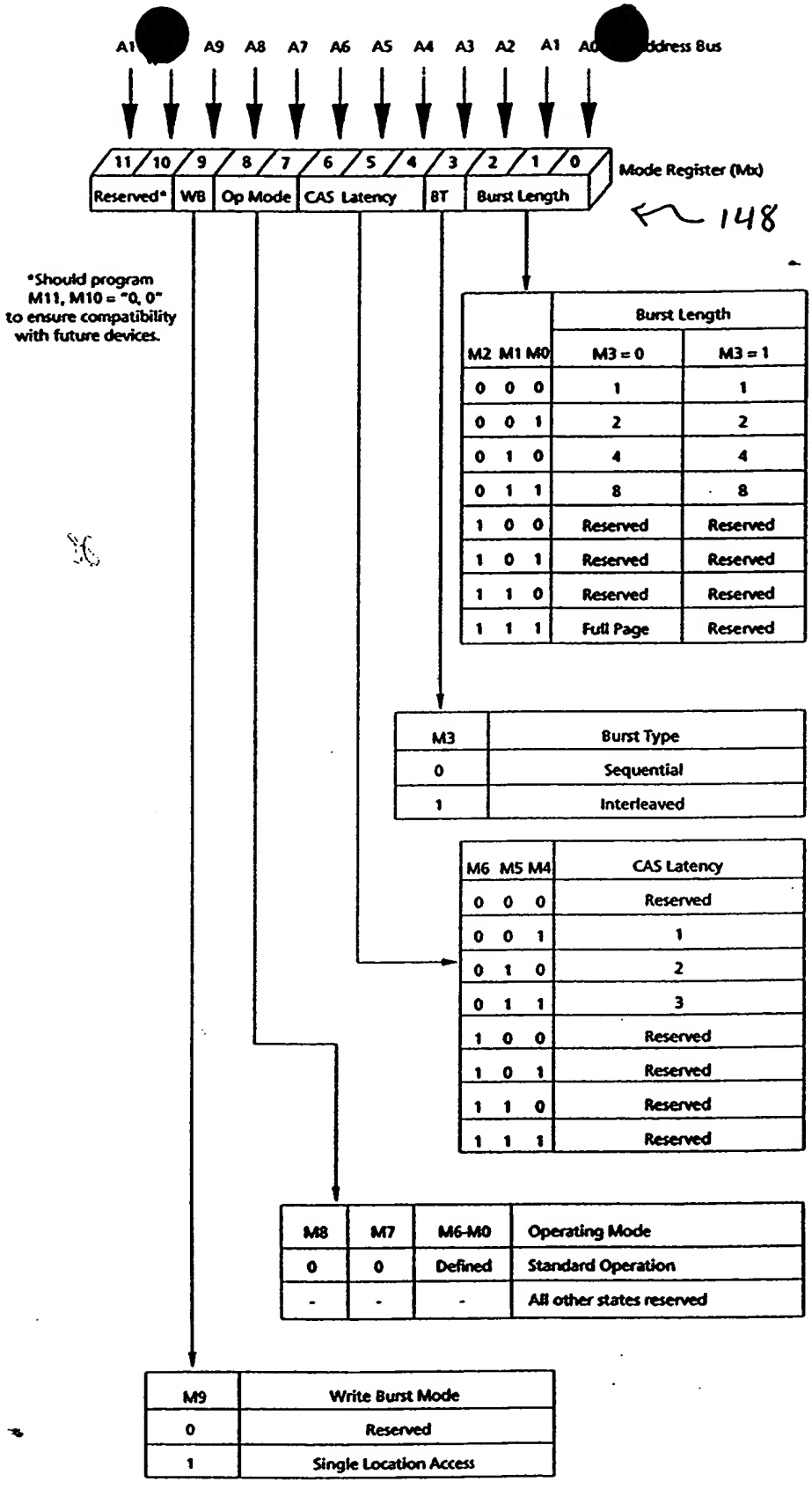

A1
A9 A8 A7 A6 A5 A4 A3 A2 A1 A0
ddress Bus
11 10 9 8 7 6 5 4 3 2 1 0
Reserved* WB Op Mode CAS Latency BT Burst Length
Mode Register (Mx)
148
*Should program M11, M10 = "0, 0" to ensure compatibility with future devices.
Burst Length
M2 M1 M0 M3 = 0 M3 = 1
0 0 0 1 1
0 0 1 2 2
0 1 0 4 4
0 1 1 8 8
1 0 0 Reserved Reserved
1 0 1 Reserved Reserved
1 1 0 Reserved Reserved
1 1 1 Full Page Reserved
M3 Burst Type
0 Sequential
1 Interleaved
M6 M5 M4 CAS Latency
0 0 0 Reserved
0 0 1 1
0 1 0 2
0 1 1 3
1 0 0 Reserved
1 0 1 Reserved
1 1 0 Reserved
1 1 1 Reserved
M8 M7 M6-M0 Operating Mode
0 0 Defined Standard Operation
- - - All other states reserved
M9 Write Burst Mode
0 Reserved
1 Single Location Access

Fig. 2

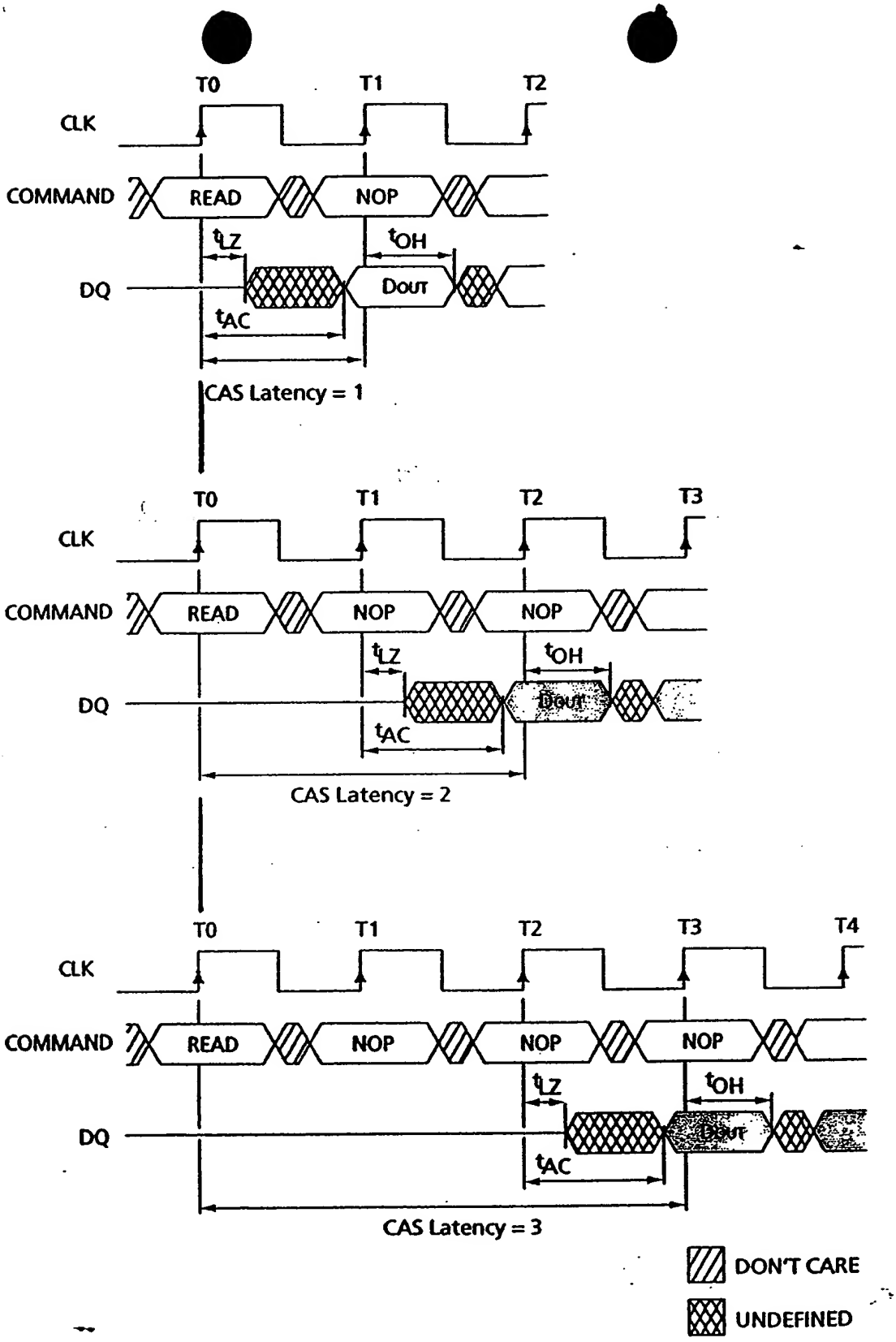

T0
T1
T2
CLK
COMMAND
READ
NOP
tLZ
tOH
DQ
DOUT
tAC
CAS Latency = 1
T0
T1
T2
T3
CLK
COMMAND
READ
NOP
NOP
tLZ
tOH
DQ
DOUT
tAC
CAS Latency = 2
T0
T1
T2
T3
T4
CLK
COMMAND
READ
NOP
NOP
NOP
tLZ
tOH
DQ
DOUT
tAC
CAS Latency = 3
DON'T CARE
UNDEFINED

Fig. 3

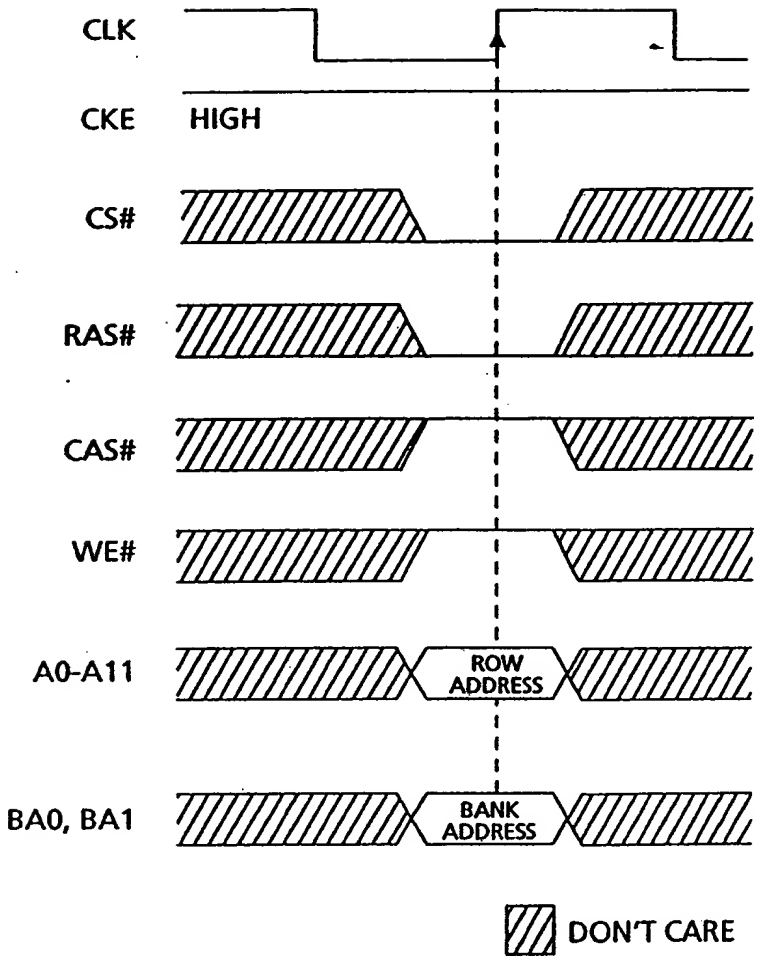

CLK
CKE
HIGH
CS#
RAS#
CAS#
WE#
A0-A11
ROW ADDRESS
BA0, BA1
BANK ADDRESS
DON'T CARE

Fig. 4

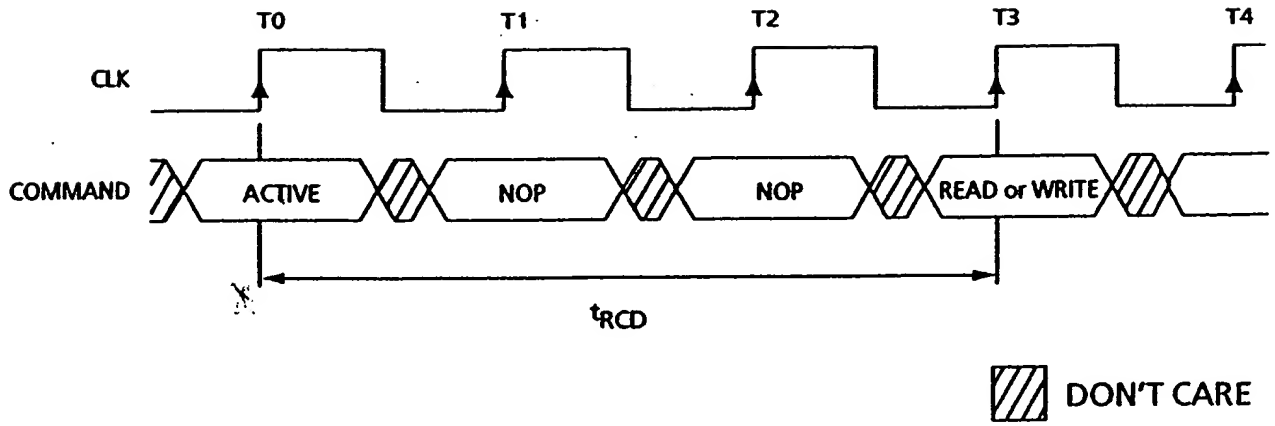

T0
T1
T2
T3
T4
CLK
COMMAND
ACTIVE
NOP
NOP
READ or WRITE
tRCD
DON'T CARE

Fig. 5

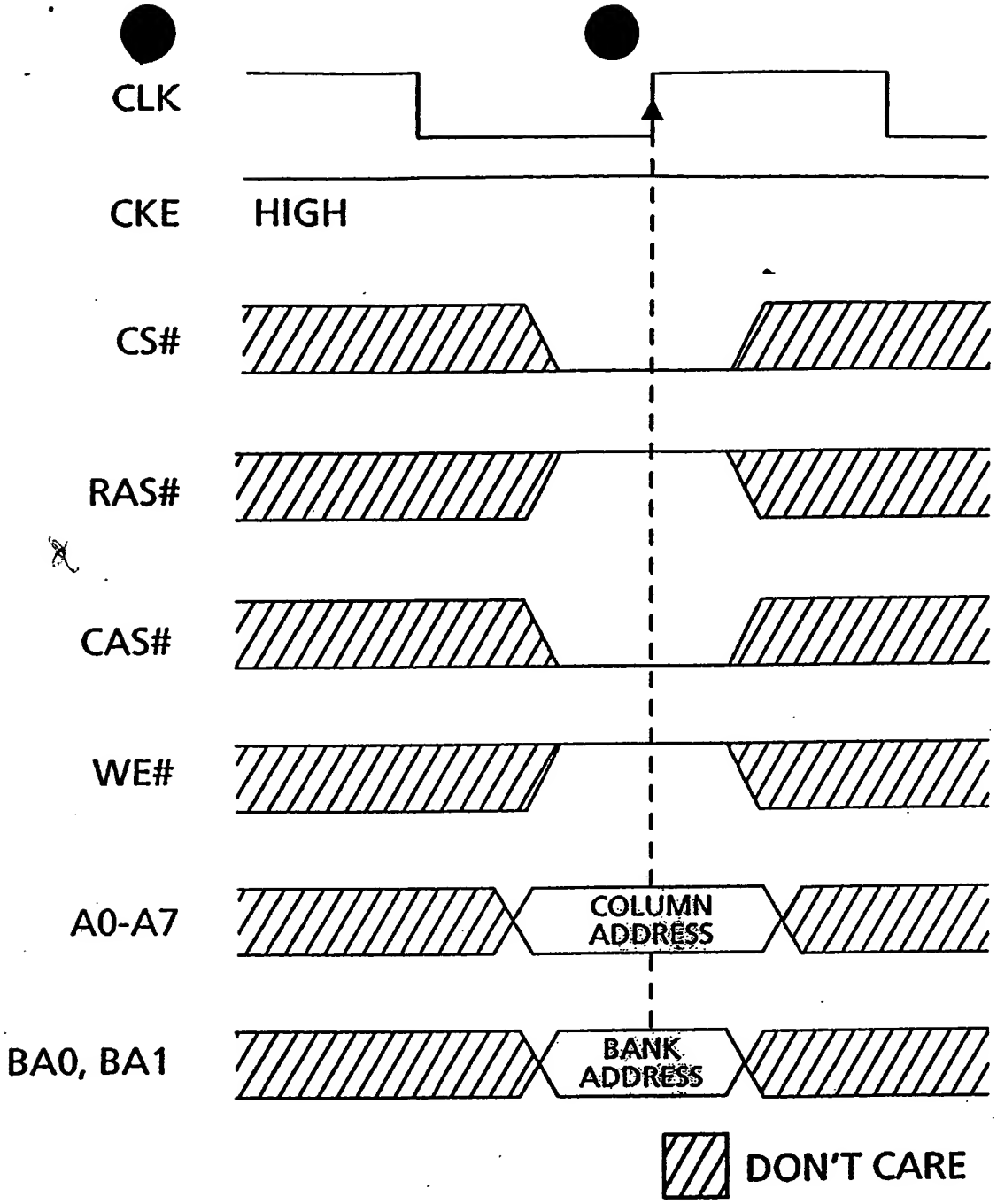
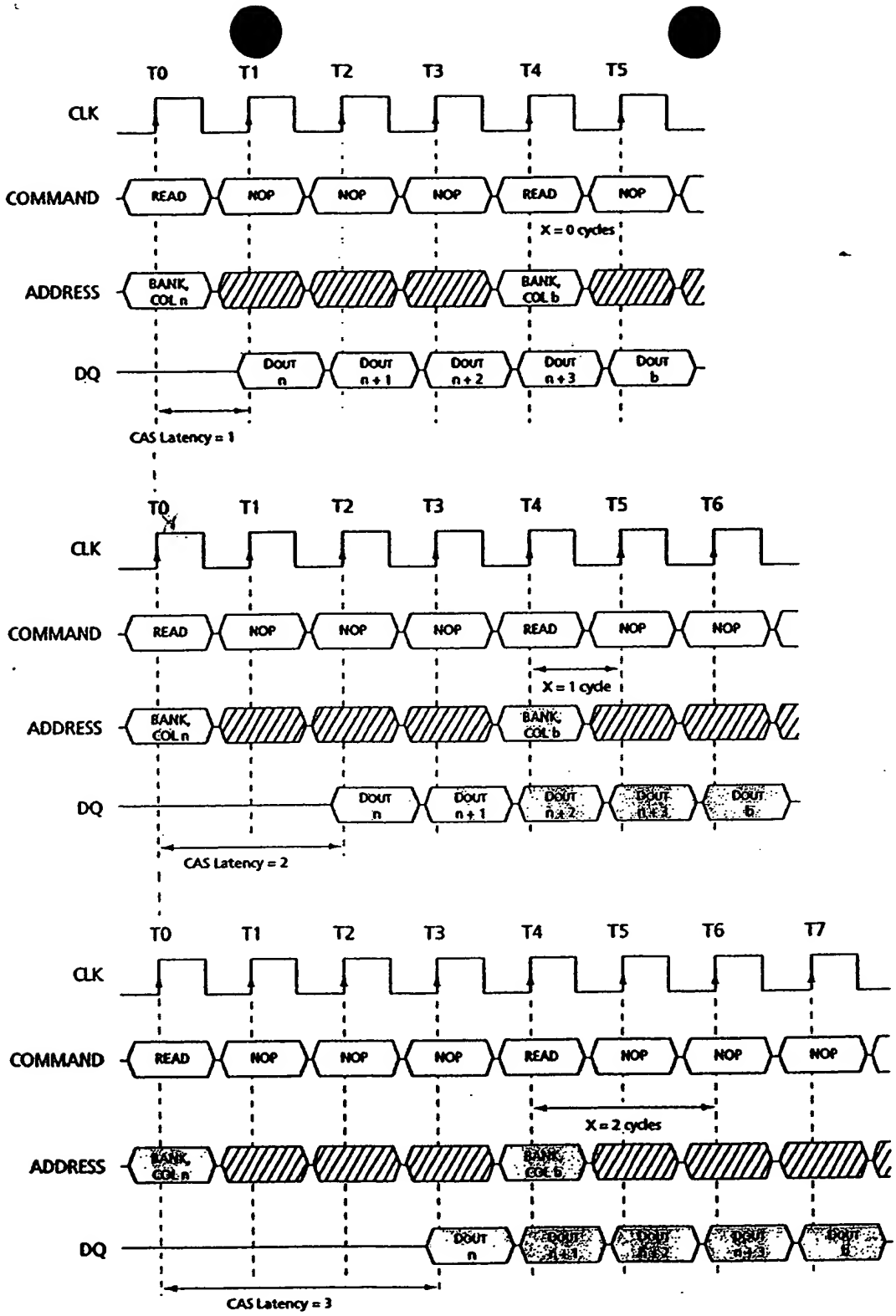
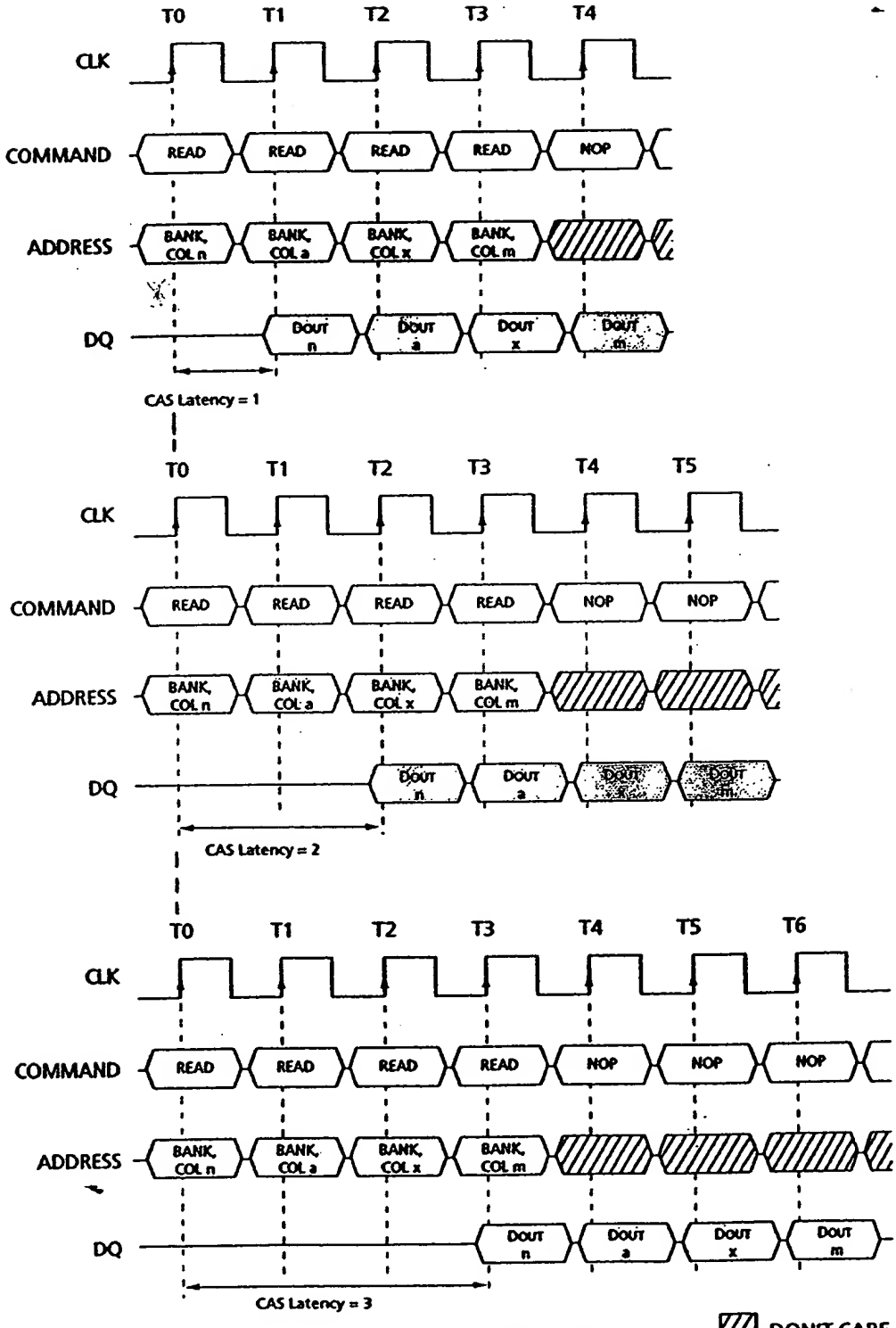

CLK
CKE
HIGH
CS#
RAS#
CAS#
WE#
A0-A7
COLUMN ADDRESS
BA0, BA1
BANK ADDRESS
DON'T CARE

Fig. 6



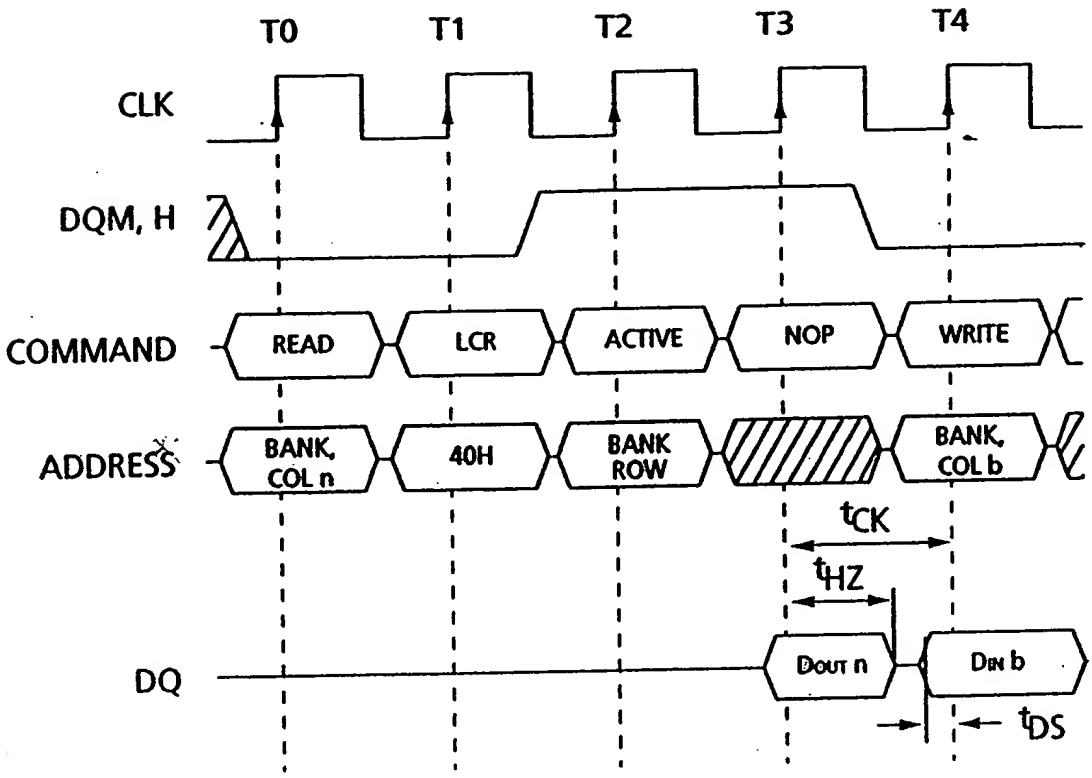
NOTE: Each READ command may be to either bank. DQM is LOW.

Fig. 7



NOTE: Each READ command may be to either bank. DQM is LOW.

Fig. 8



NOTE: A CAS latency of three is used for illustration. The READ command may be to any bank, and the WRITE command may be to any bank. If a CAS latency of one is used, then DQM is not required.

DON'T CARE

Fig 9

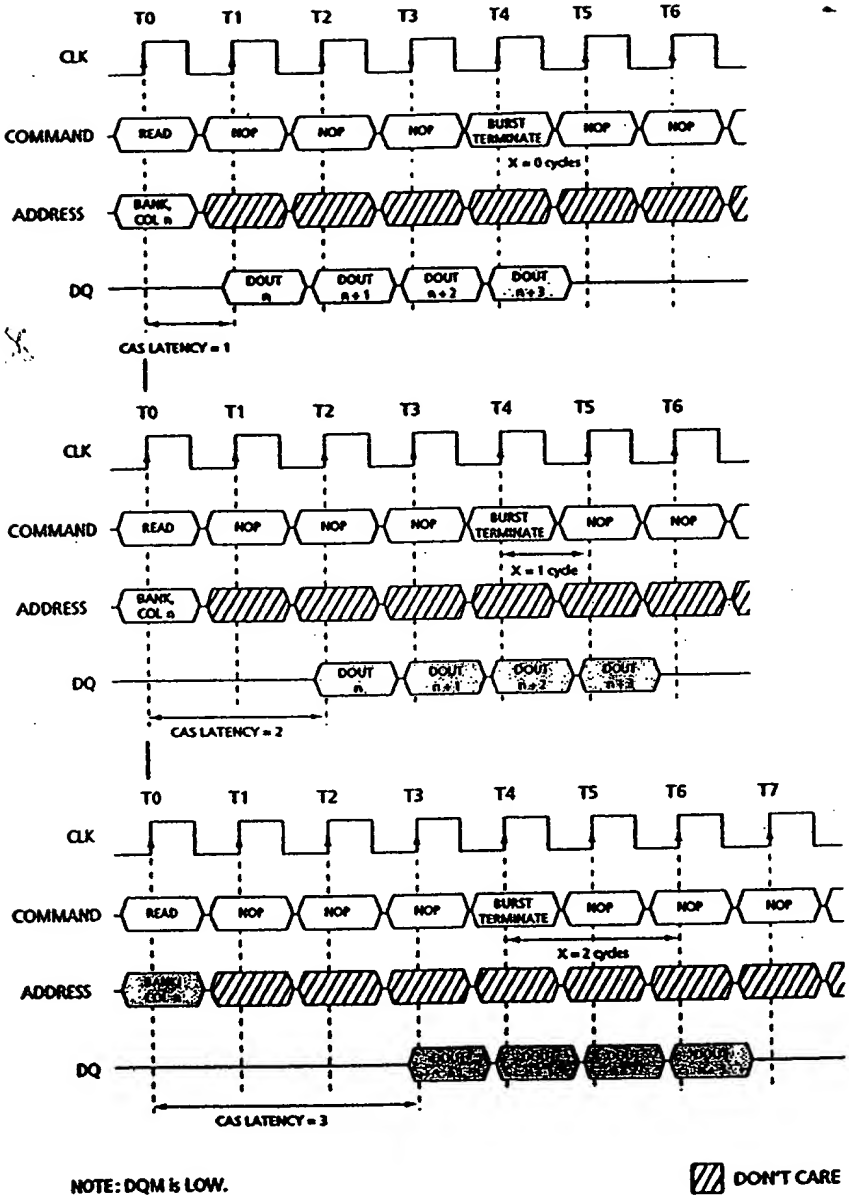

T0
T1
T2
T3
T4
T5
T6
CLK
COMMAND
READ
NOP
BURST TERMINATE
X = 0 cycles
ADDRESS
BANK, COL n
DQ
DOUT n
DOUT n+1
DOUT n+2
DOUT n+3
CAS LATENCY = 1
X = 1 cycle
DOUT n
DOUT n+1
DOUT n+2
DOUT n+3
CAS LATENCY = 2
T7
X = 2 cycles
CAS LATENCY = 3
NOTE: DQM is LOW.
DON'T CARE

Fig. 10

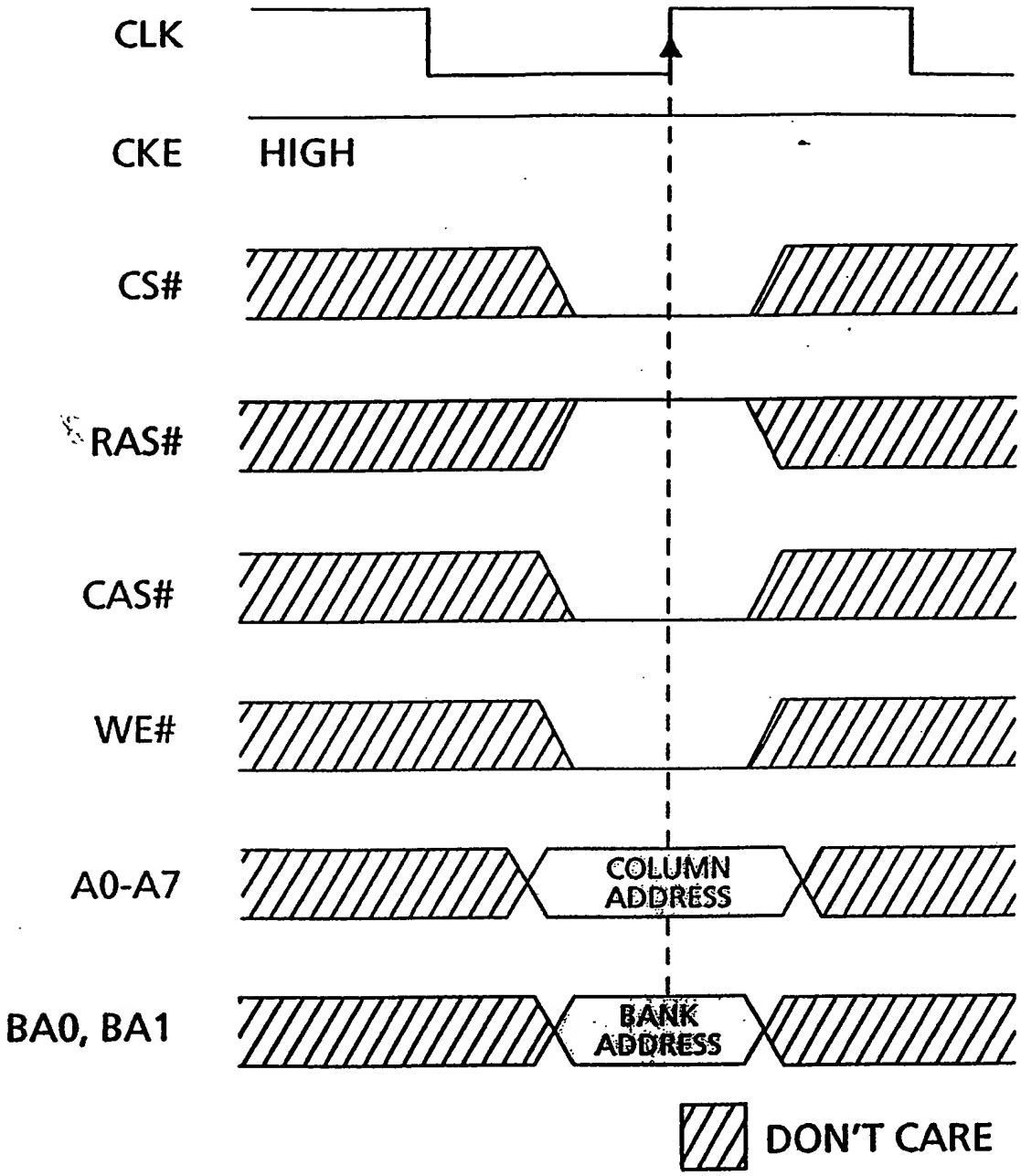
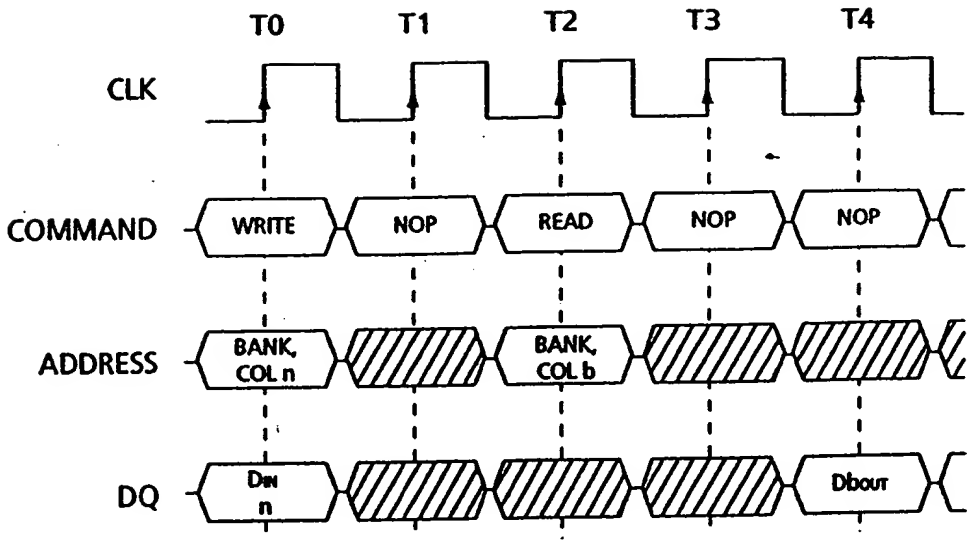

CLK
CKE
HIGH
CS#
RAS#
CAS#
WE#
A0-A7
COLUMN ADDRESS
BA0, BA1
BANK ADDRESS
DON'T CARE

Fig. 11



NOTE: A CAS latency of two is used for illustration. The WRITE command may be to any bank and the READ command may be to any bank. DQM is LOW. A READ to the bank undergoing the WRITE ISM operation may output invalid data.

▨ DON'T CARE

Fig. 12

Coming out of a power-down sequence (active),
^{t}CKS (CKE setup time) must be greater than or equal to 3ns.

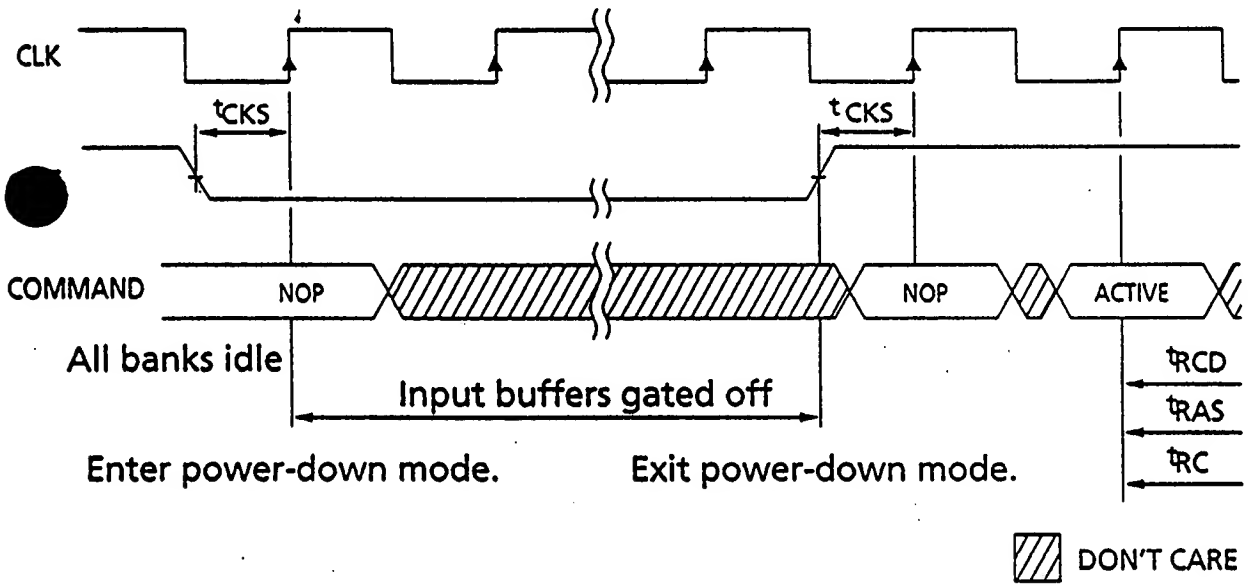
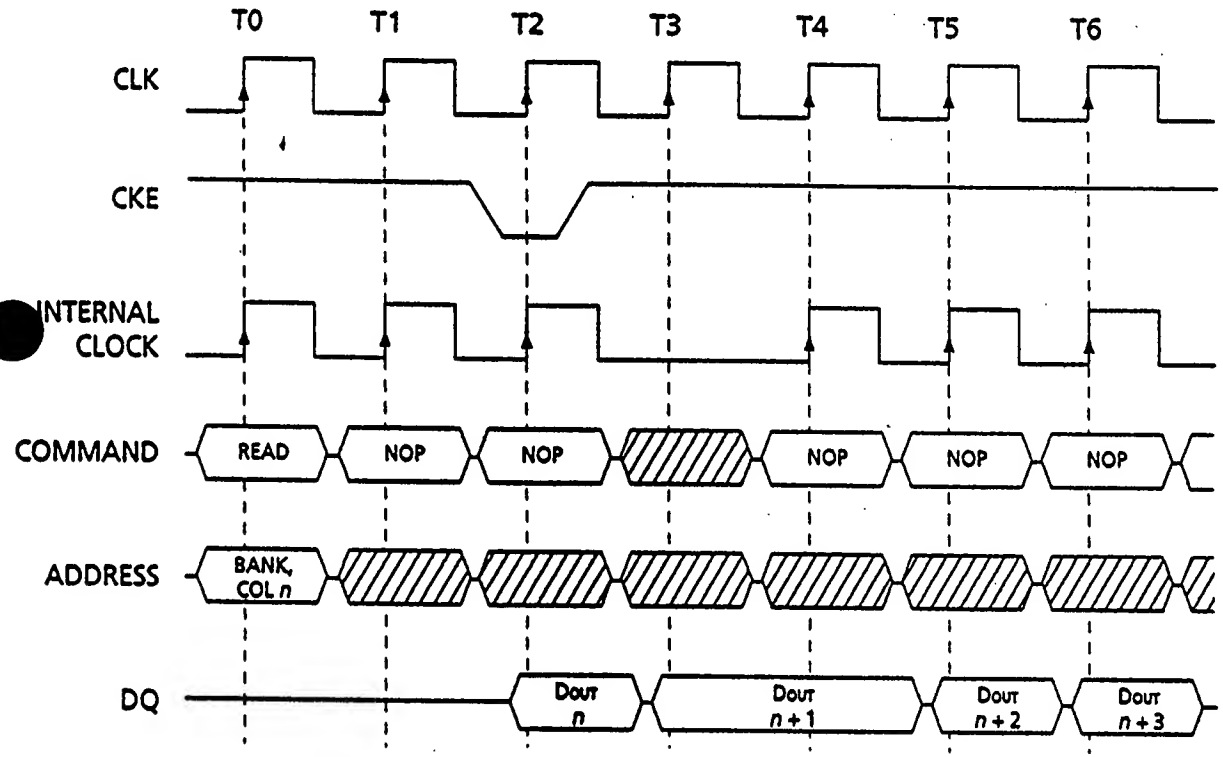


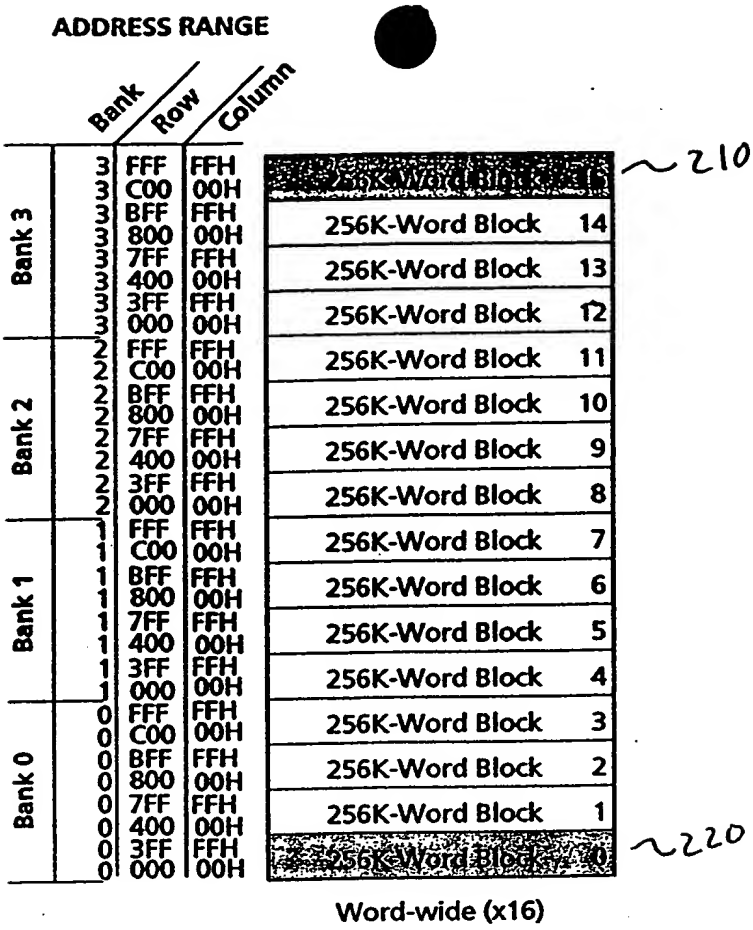
Fig. 13



NOTE: For this example, CAS latency = 2, burst length = 4 or greater, an
DQM is LOW.

▨ DON'T CARE

Fig. 14



Word-wide (x16)

Software Lock = Hardware-Lock Sectors
RP# = V_{HH} to unprotect if either the block protect or device protect bit is set.

Software Lock = Hardware-Lock Sectors
RP# = V_{CC} to unprotect but must be V_{HH} if the device protect bit is set.

See BLOCK PROTECT/UNPROTECT SEQUENCE for detailed information.

Fig. 15

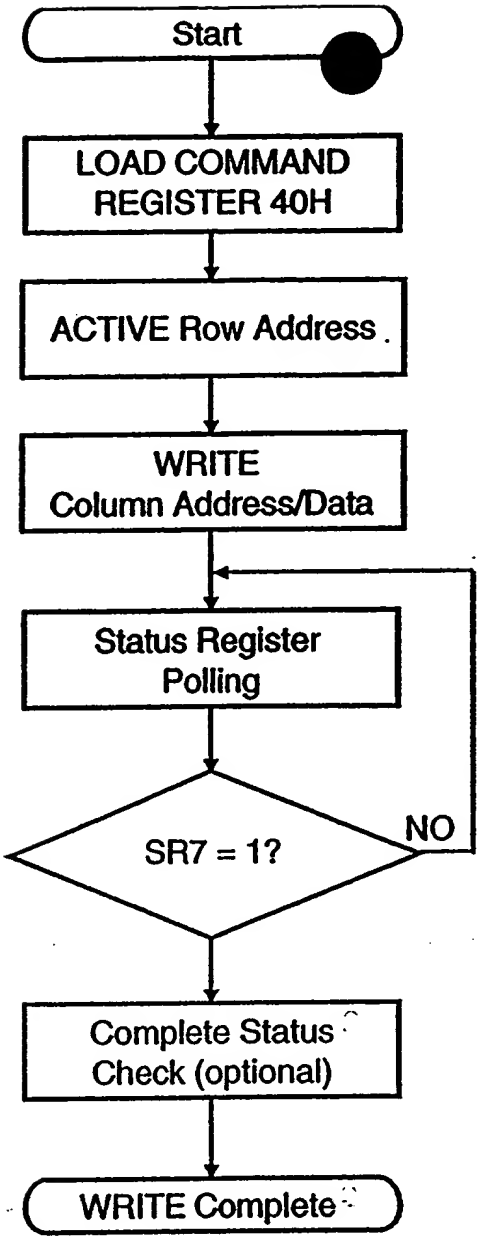

Start
LOAD COMMAND REGISTER 40H
ACTIVE Row Address
WRITE Column Address/Data
Status Register Polling
SR7 = 1?
NO
Complete Status Check (optional)
WRITE Complete

Fig. 16

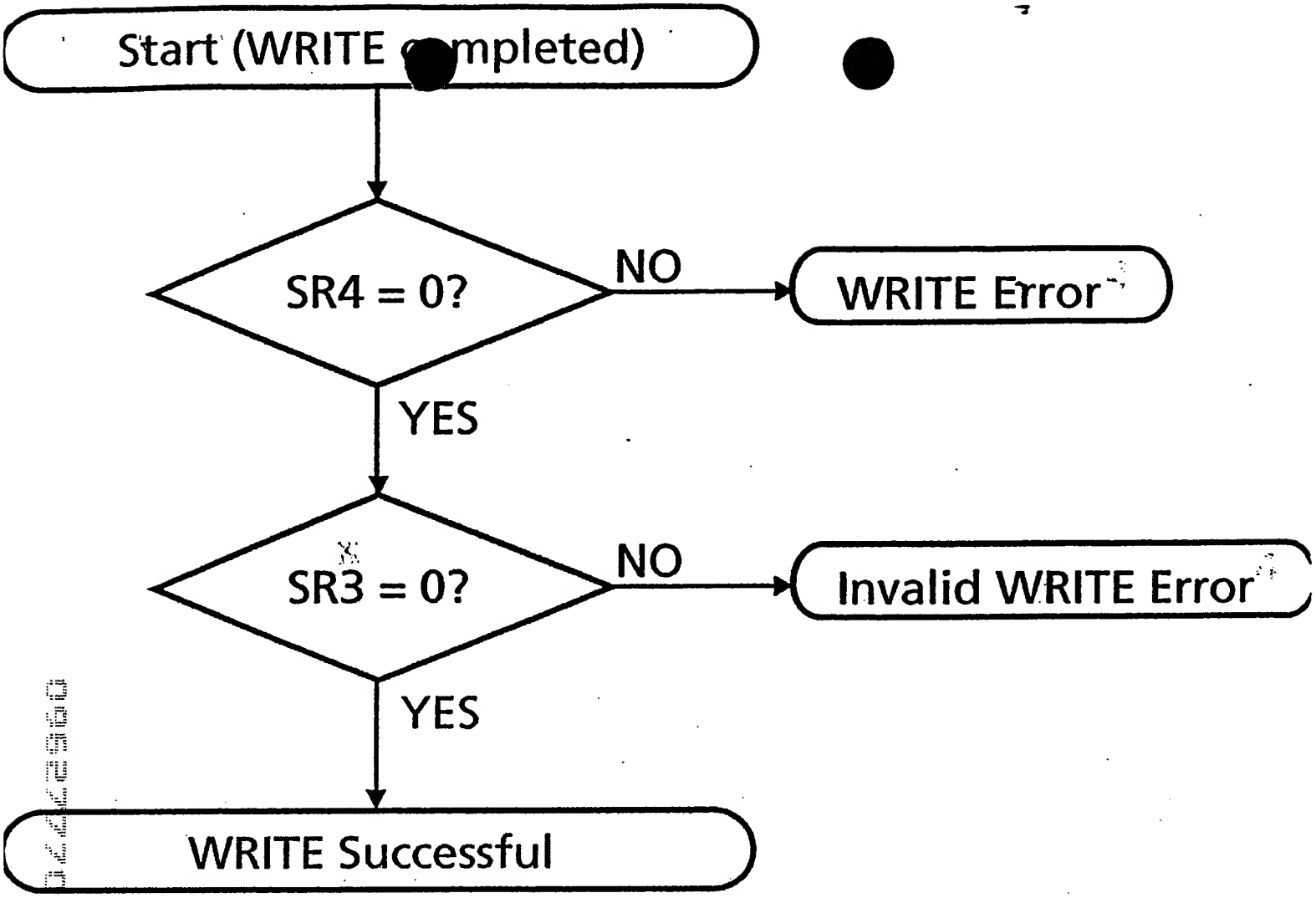

Start (WRITE completed)
SR4 = 0?
NO
WRITE Error
YES
SR3 = 0?
NO
Invalid WRITE Error
YES
WRITE Successful

Fig. 17

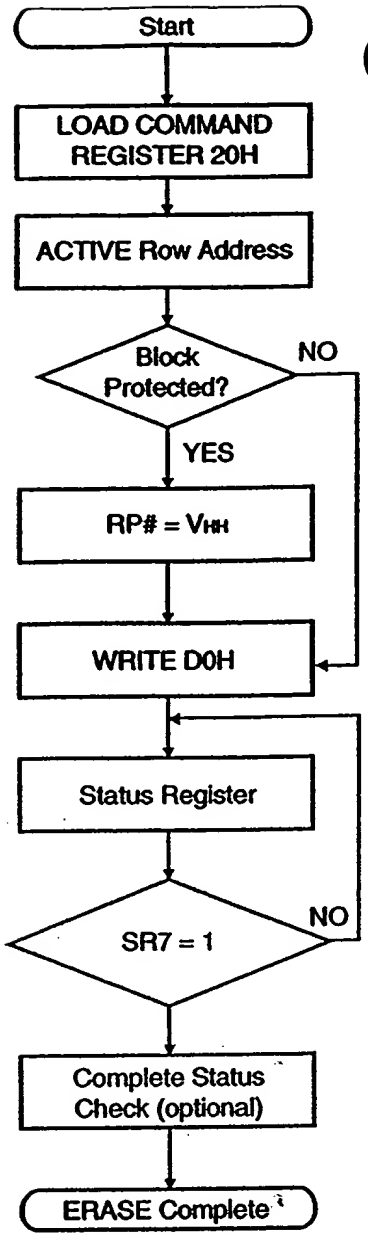

Start
LOAD COMMAND REGISTER 20H
ACTIVE Row Address
Block Protected?
NO
YES
RP# = VHH
WRITE D0H
Status Register
SR7 = 1
NO
Complete Status Check (optional)
ERASE Complete

Fig. 18

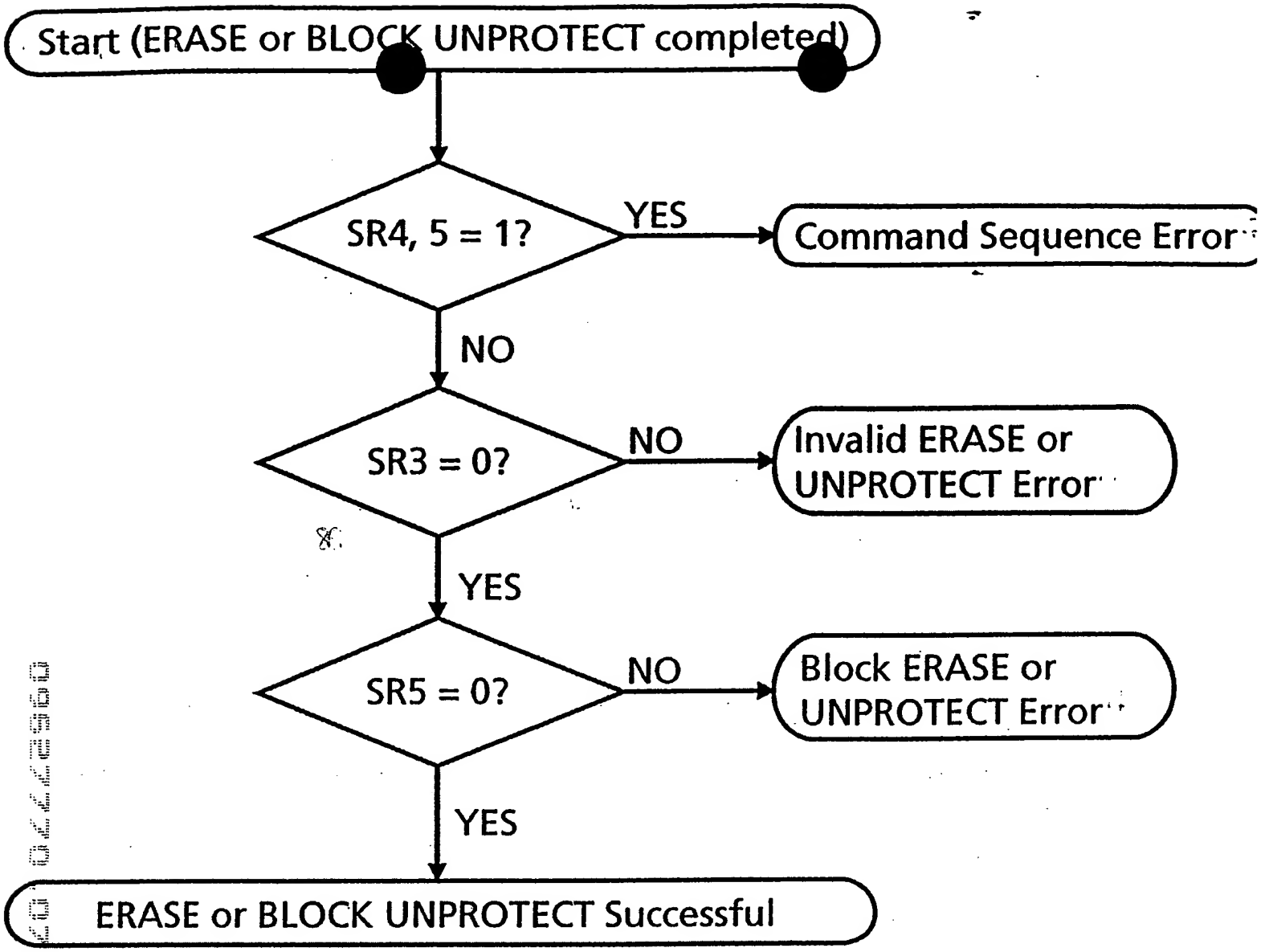

Start (ERASE or BLOCK UNPROTECT completed)
SR4, 5 = 1?
YES
Command Sequence Error
NO
SR3 = 0?
NO
Invalid ERASE or UNPROTECT Error
YES
SR5 = 0?
NO
Block ERASE or UNPROTECT Error
YES
ERASE or BLOCK UNPROTECT Successful

Fig. 19

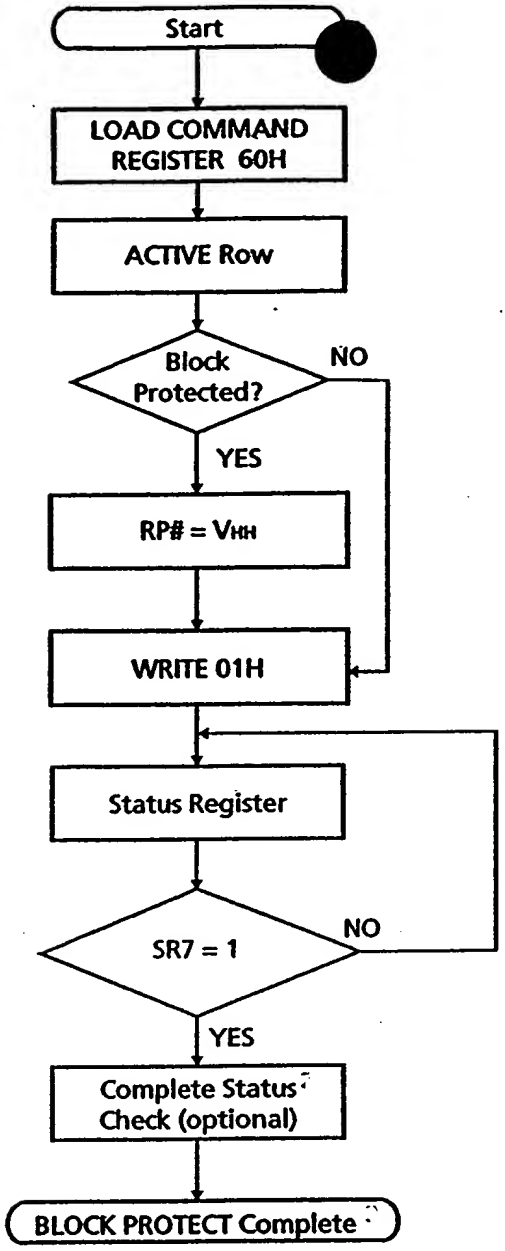

Start
LOAD COMMAND REGISTER 60H
ACTIVE Row
Block Protected?
NO
YES
RP# = VHH
WRITE 01H
Status Register
SR7 = 1
NO
YES
Complete Status Check (optional)
BLOCK PROTECT Complete

Fig. 20

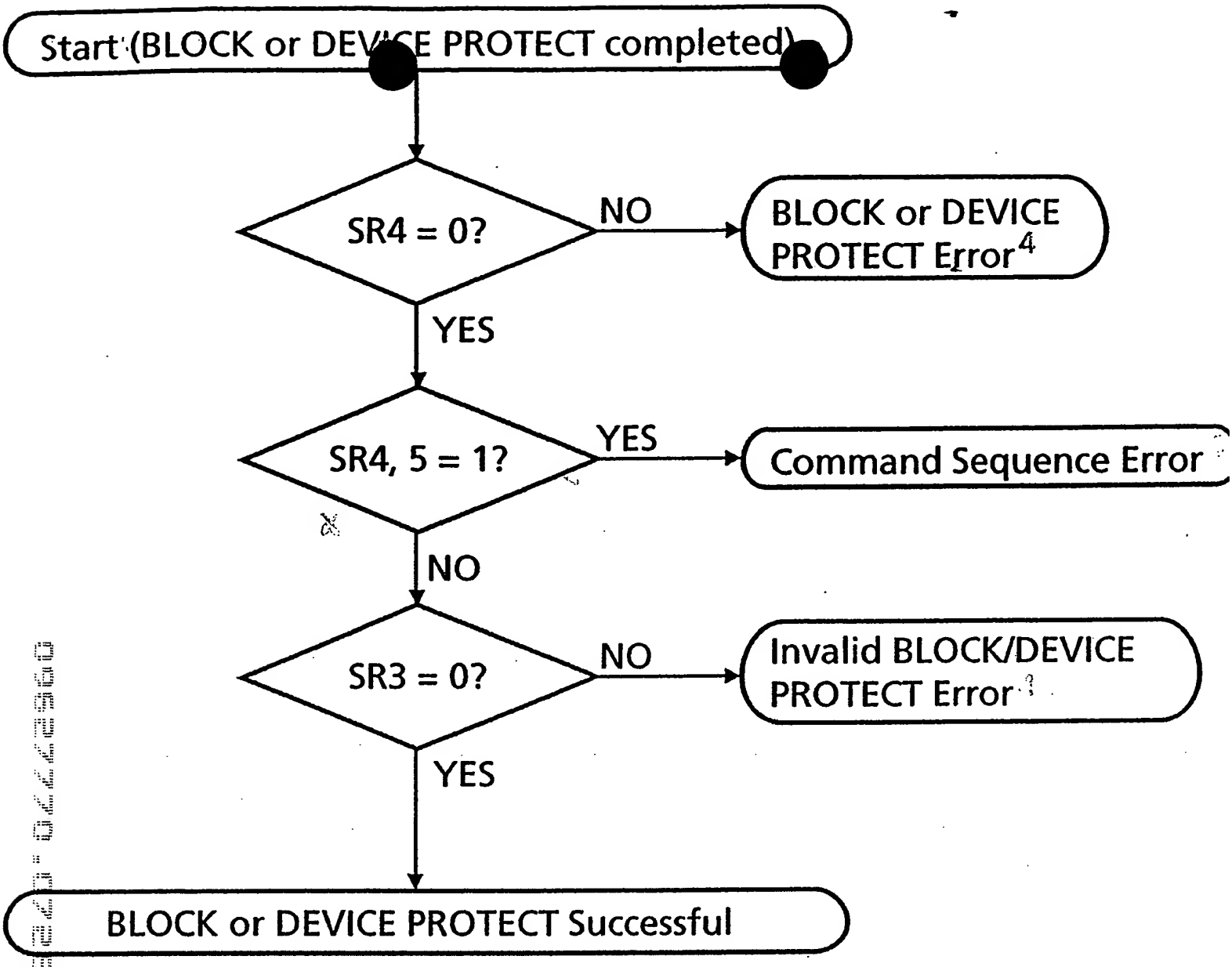

Start (BLOCK or DEVICE PROTECT completed)
SR4 = 0?
NO
BLOCK or DEVICE PROTECT Error4
YES
SR4, 5 = 1?
YES
Command Sequence Error
NO
SR3 = 0?
NO
Invalid BLOCK/DEVICE PROTECT Error3
YES
BLOCK or DEVICE PROTECT Successful

Fig. 21

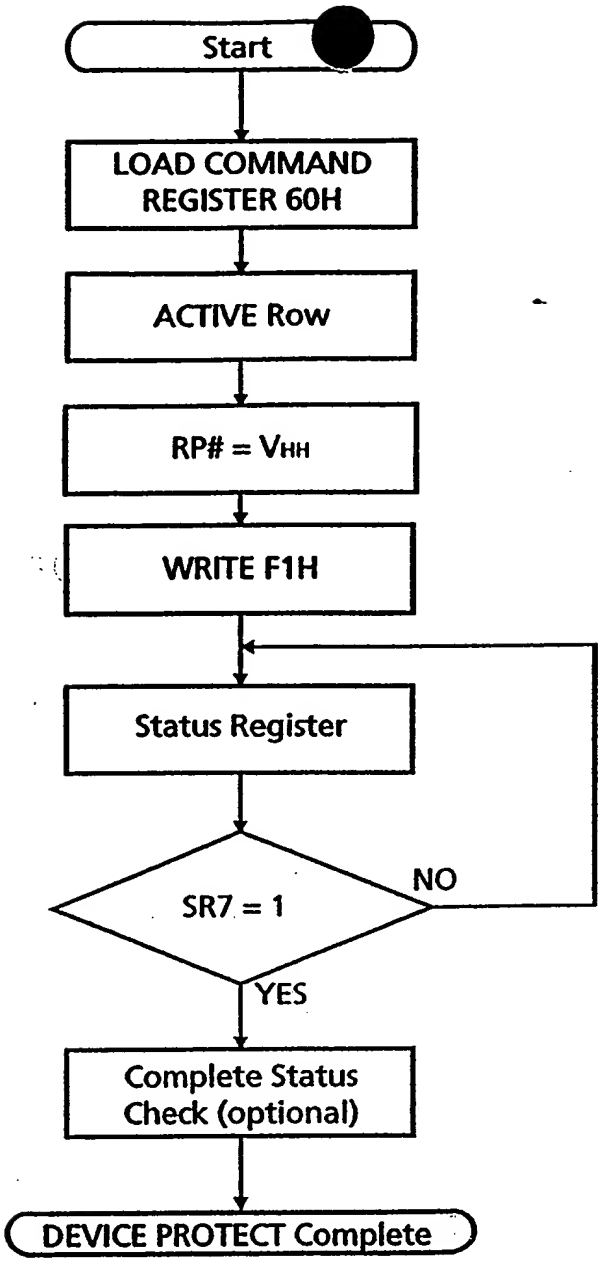

Start
LOAD COMMAND REGISTER 60H
ACTIVE Row
RP# = VHH
WRITE F1H
Status Register
SR7 = 1
NO
YES
Complete Status Check (optional)
DEVICE PROTECT Complete

Fig. 22

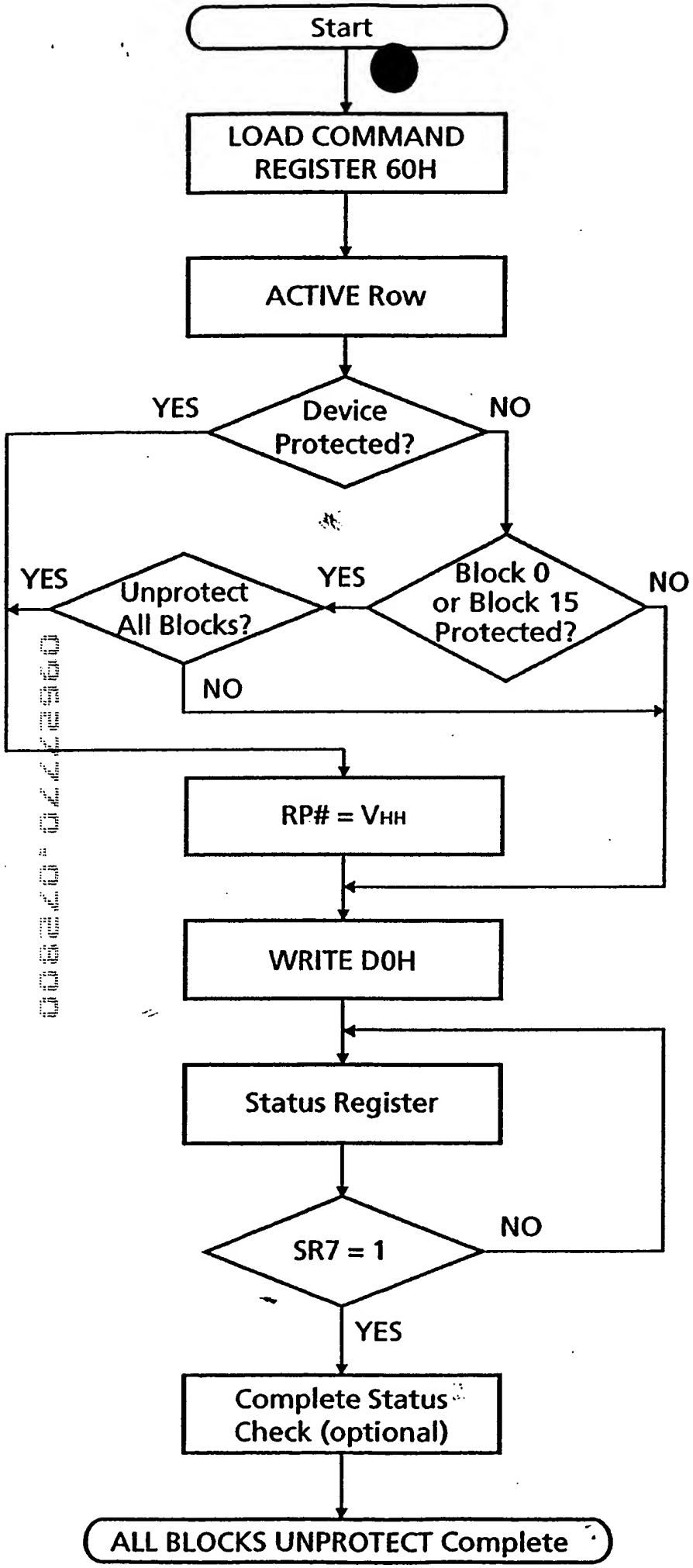

Start
LOAD COMMAND REGISTER 60H
ACTIVE Row
YES
Device Protected?
NO
YES
Unprotect All Blocks?
YES
Block 0 or Block 15 Protected?
NO
NO
RP# = VHH
WRITE D0H
Status Register
SR7 = 1
NO
YES
Complete Status Check (optional)
ALL BLOCKS UNPROTECT Complete

Fig. 23

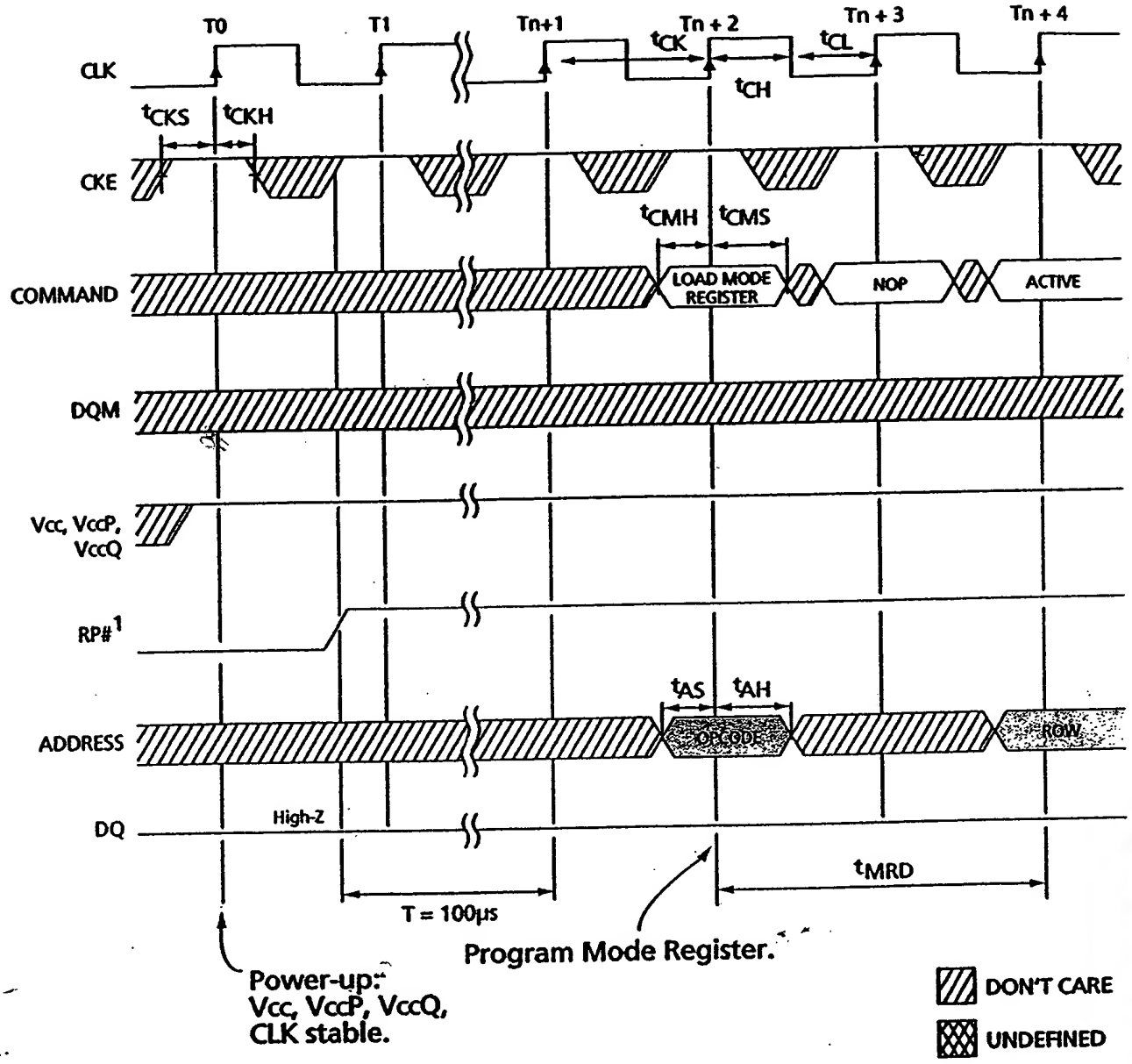

T0
T1
Tn+1
Tn + 2
Tn + 3
Tn + 4
CLK
tCK
tCL
tCH
tCKS
tCKH
CKE
tCMH
tCMS
COMMAND
LOAD MODE REGISTER
NOP
ACTIVE
DQM
Vcc, VccP, VccQ
RP#1
tAS
tAH
ADDRESS
OPCODE
ROW
DQ
High-Z
tMRD
T = 100µs
Program Mode Register.
Power-up:
Vcc, VccP, VccQ,
CLK stable.
DON'T CARE
UNDEFINED

Fig. 24

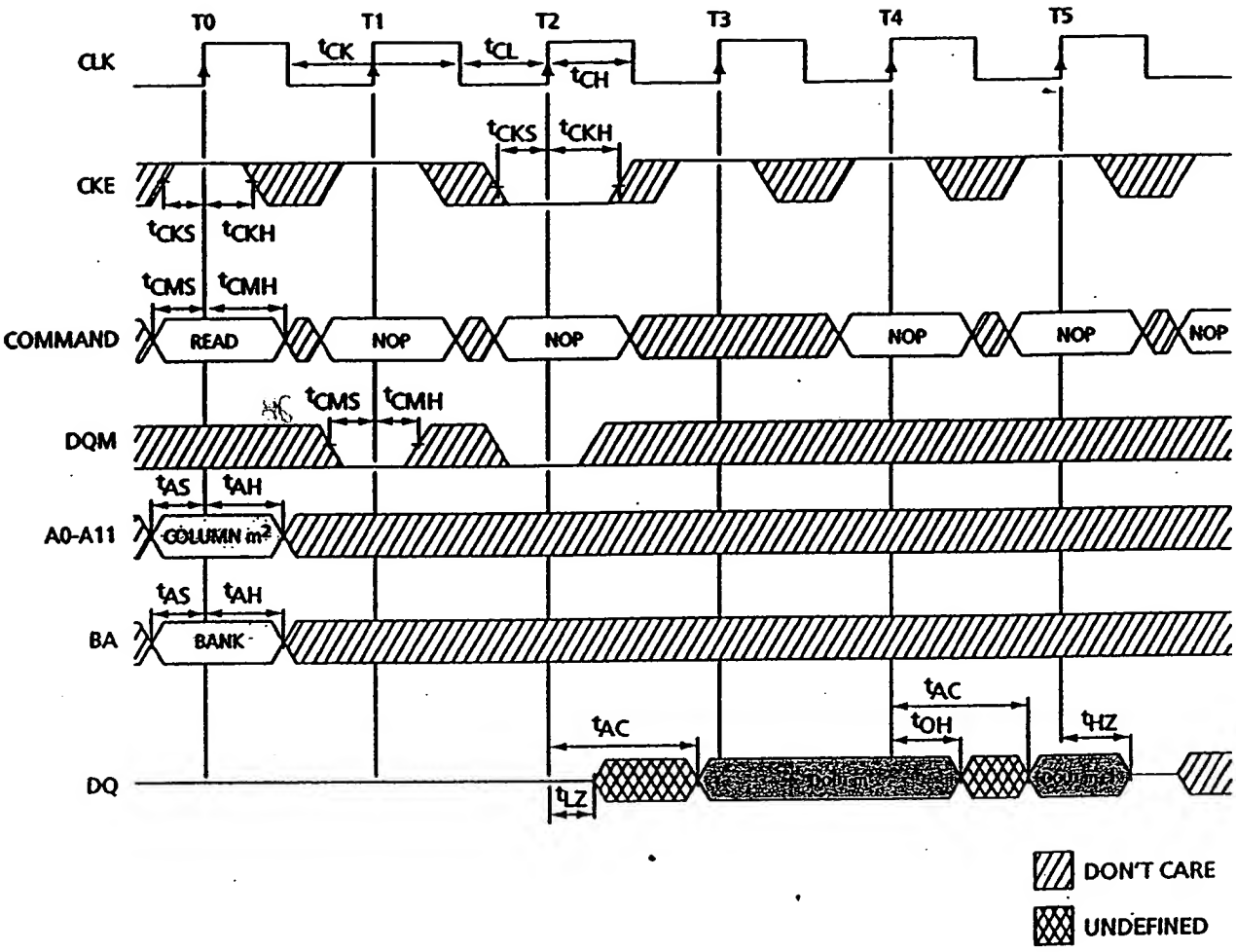

T0
T1
T2
T3
T4
T5
CLK
tCK
tCL
tCH
CKE
tCKS
tCKH
tCMS
tCMH
COMMAND
READ
NOP
NOP
NOP
NOP
NOP
DQM
tAS
tAH
A0-A11
COLUMN m
BA
BANK
tAC
tOH
tHZ
tLZ
DQ
DON'T CARE
UNDEFINED

Fig. 25

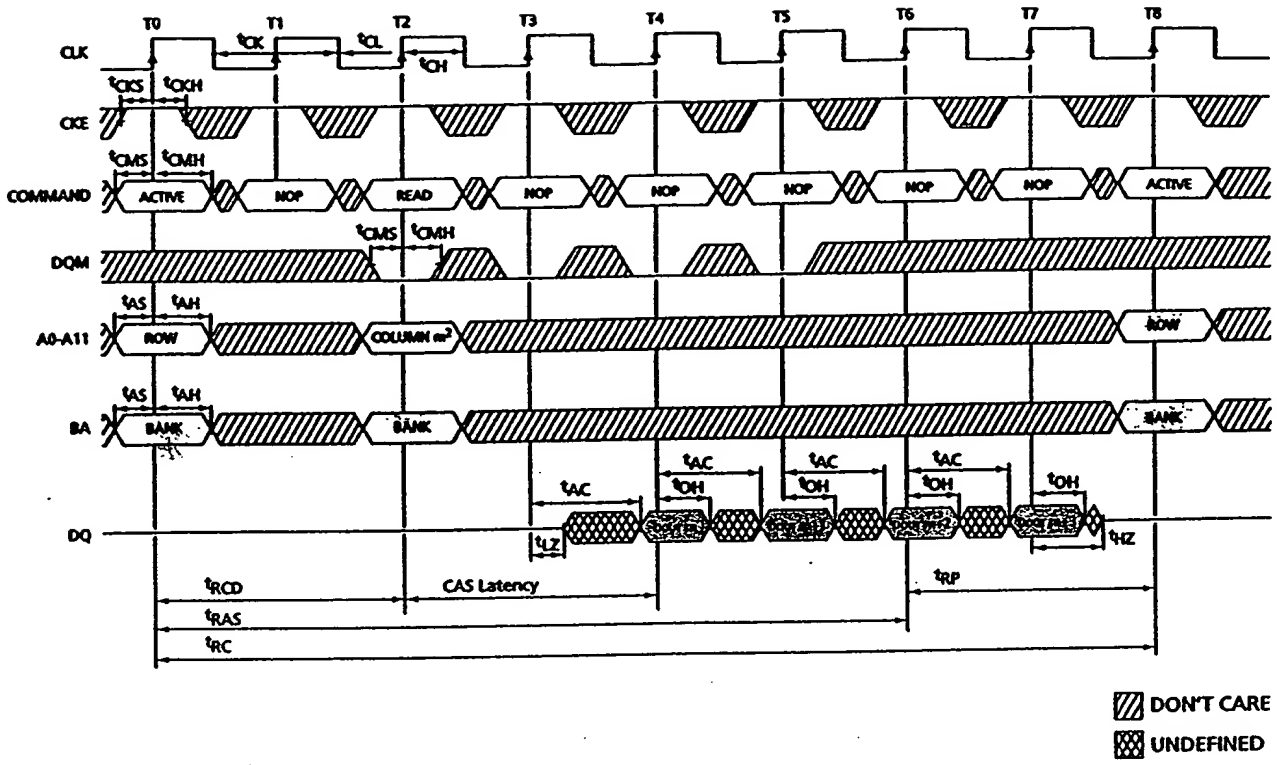

T0
T1
T2
T3
T4
T5
T6
T7
T8
CLK
CKE
COMMAND
DQM
A0-A11
BA
DQ
ACTIVE
NOP
READ
ROW
COLUMN m
BANK
CAS Latency
DON'T CARE
UNDEFINED

Fig. 26

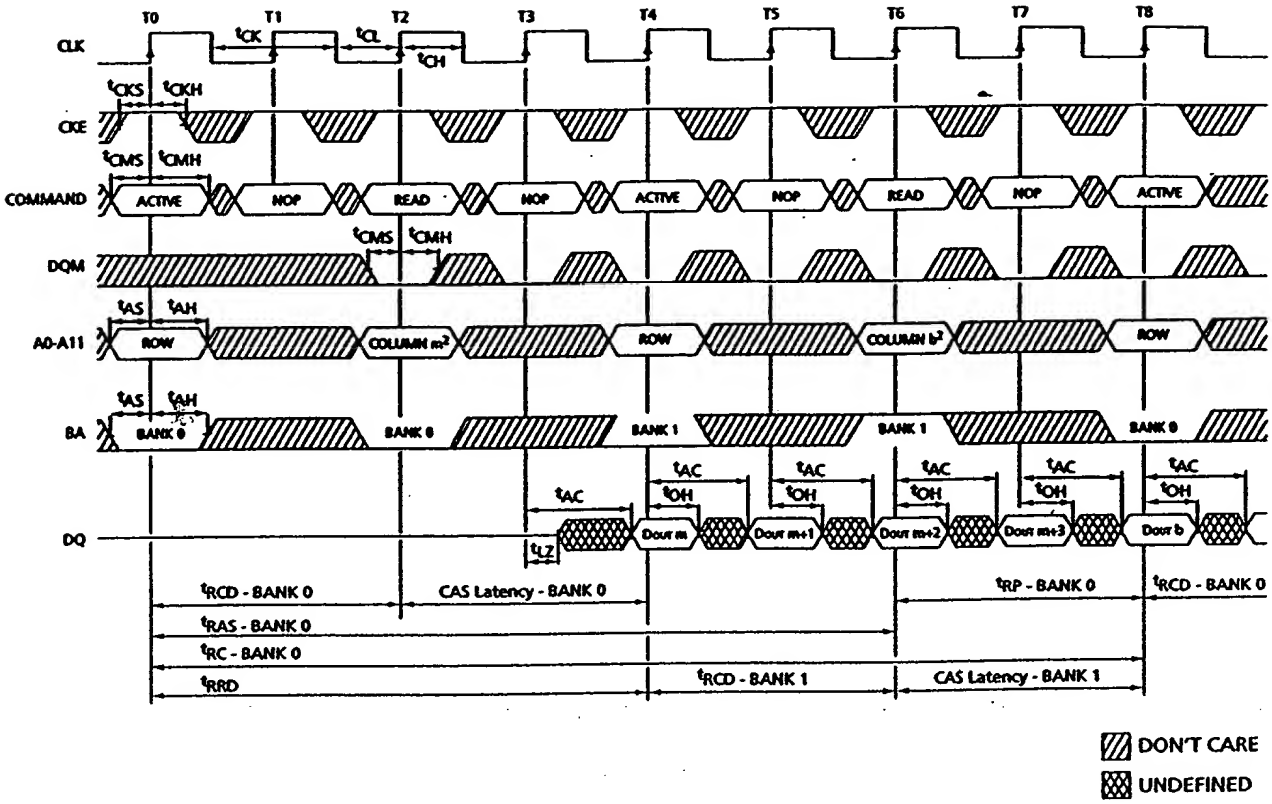

T0
T1
T2
T3
T4
T5
T6
T7
T8
CLK
CKE
COMMAND
DQM
A0-A11
BA
DQ
ACTIVE
NOP
READ
ROW
BANK 0
BANK 1
tRCD - BANK 0
CAS Latency - BANK 0
tRP - BANK 0
tRAS - BANK 0
tRC - BANK 0
tRRD
tRCD - BANK 1
CAS Latency - BANK 1
DON'T CARE
UNDEFINED

Fig. 27

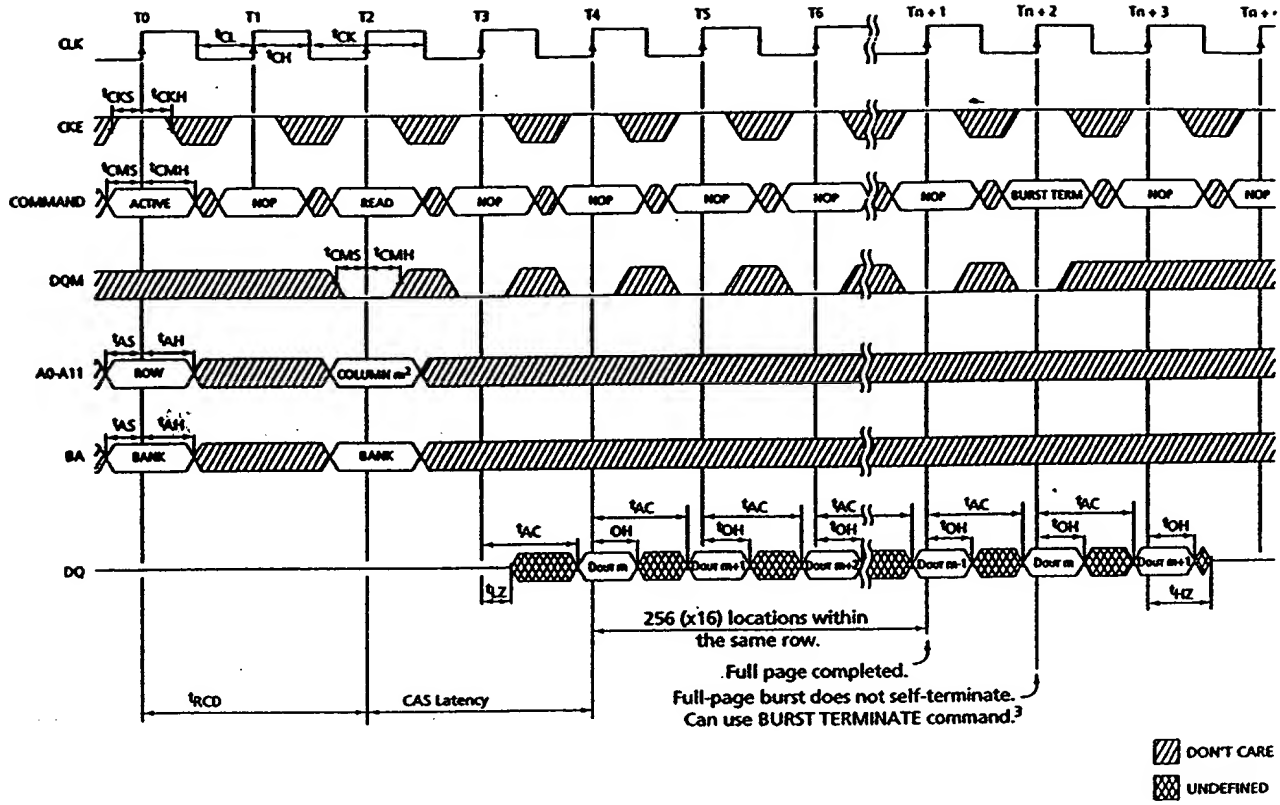
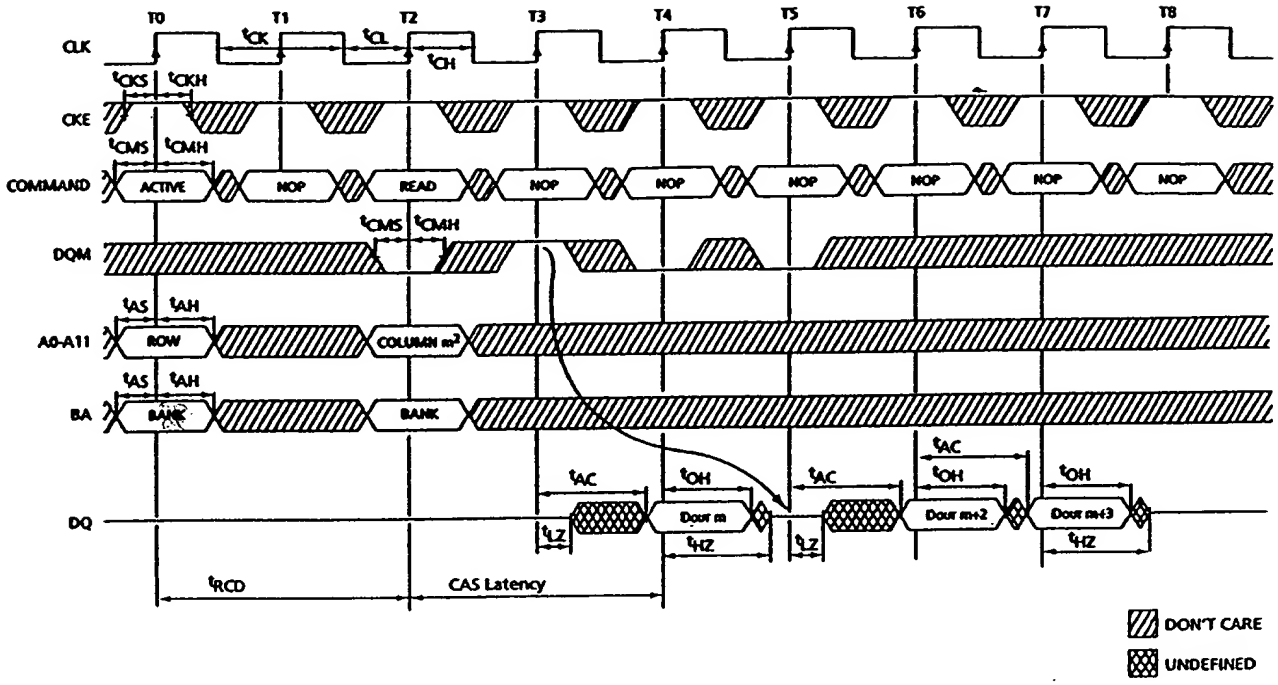

T0
T1
T2
T3
T4
T5
T6
Tn + 1
Tn + 2
Tn + 3
CLK
CKE
COMMAND
ACTIVE
NOP
READ
BURST TERM
DQM
A0-A11
ROW
COLUMN m
BA
BANK
DQ
256 (x16) locations within the same row.
Full page completed.
Full-page burst does not self-terminate.
Can use BURST TERMINATE command.
tRCD
CAS Latency
DON'T CARE
UNDEFINED

Fig. 28


T0
T1
T2
T3
T4
T5
T6
T7
T8
CLK
CKE
COMMAND
DQM
A0-A11
BA
DQ
ACTIVE
NOP
READ
ROW
COLUMN m
BANK
Dout m
Dout m+2
Dout m+3
tRCD
CAS Latency
DON'T CARE
UNDEFINED

Frg. 29

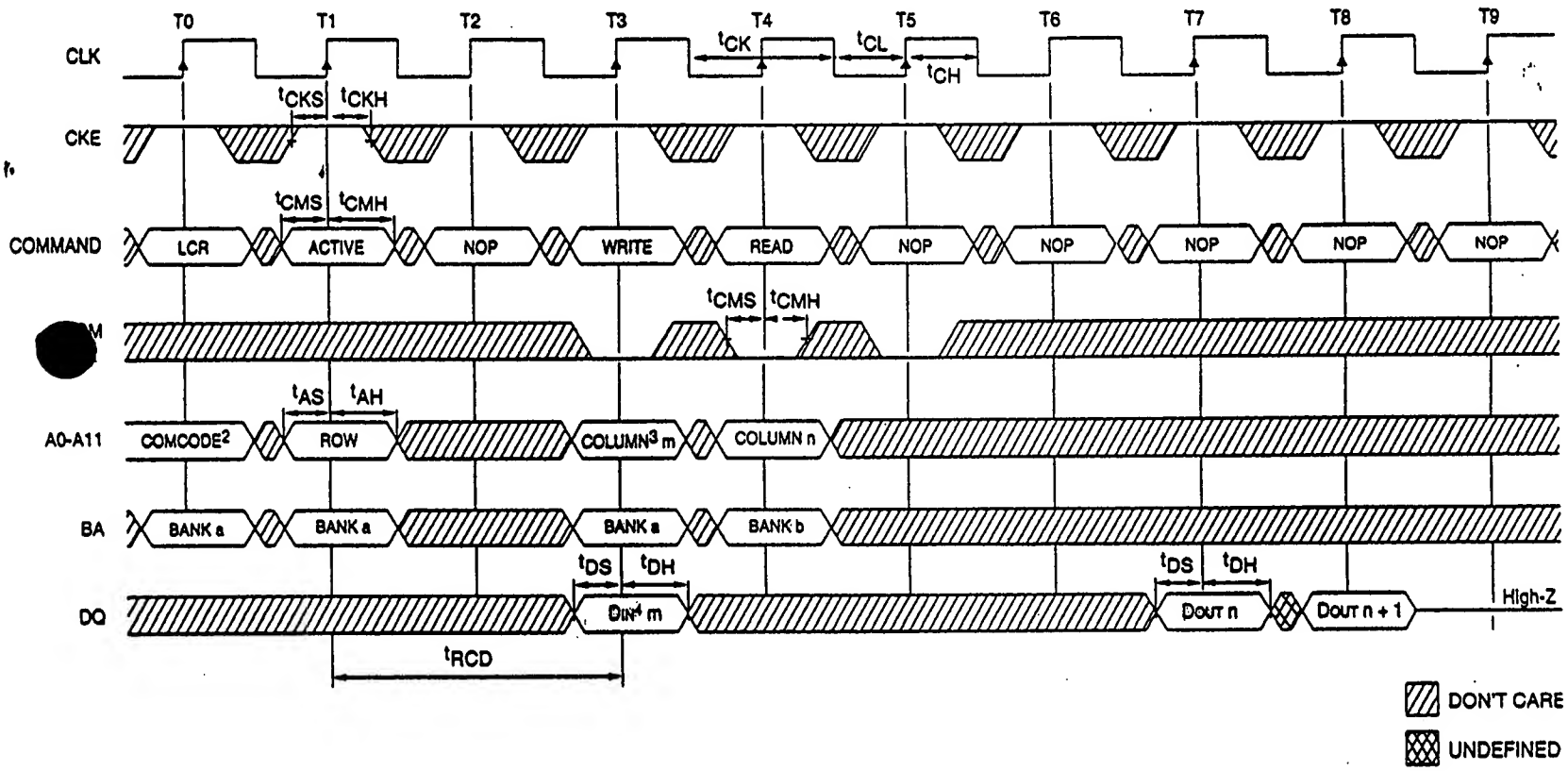


Fig. 30

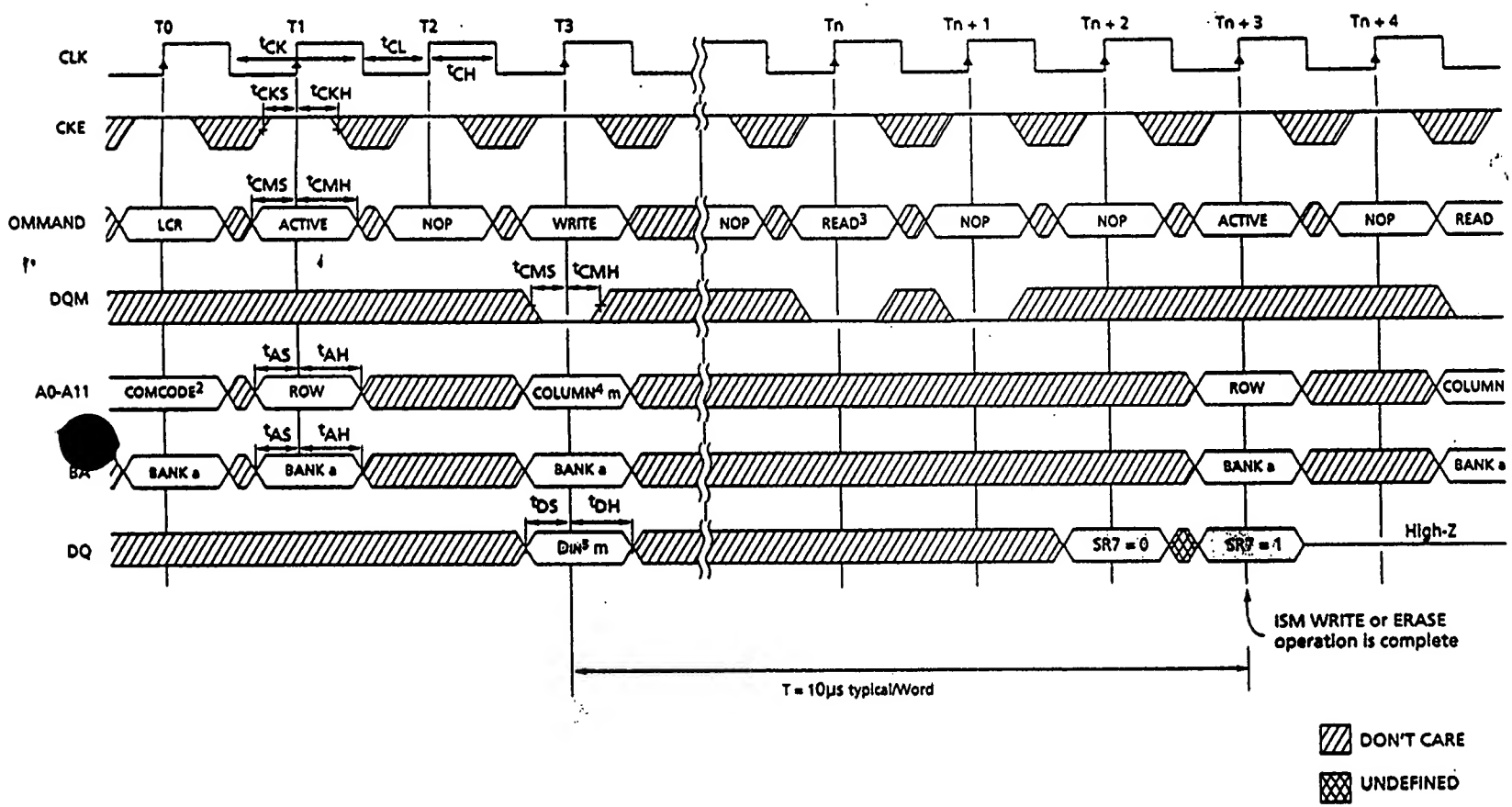

T0
T1
T2
T3
Tn
Tn + 1
Tn + 2
Tn + 3
Tn + 4
CLK
CKE
OMMAND
DQM
A0-A11
BA
DQ
tCK
tCL
tCH
tCKS
tCKH
tCMS
tCMH
tAS
tAH
tDS
tDH
LCR
ACTIVE
NOP
WRITE
READ3
READ
COMCODE2
ROW
COLUMN4 m
COLUMN
BANK a
DIN5 m
SR7 = 0
SR7 = 1
High-Z
ISM WRITE or ERASE operation is complete
T = 10µs typical/Word
DON'T CARE
UNDEFINED

Fig. 31

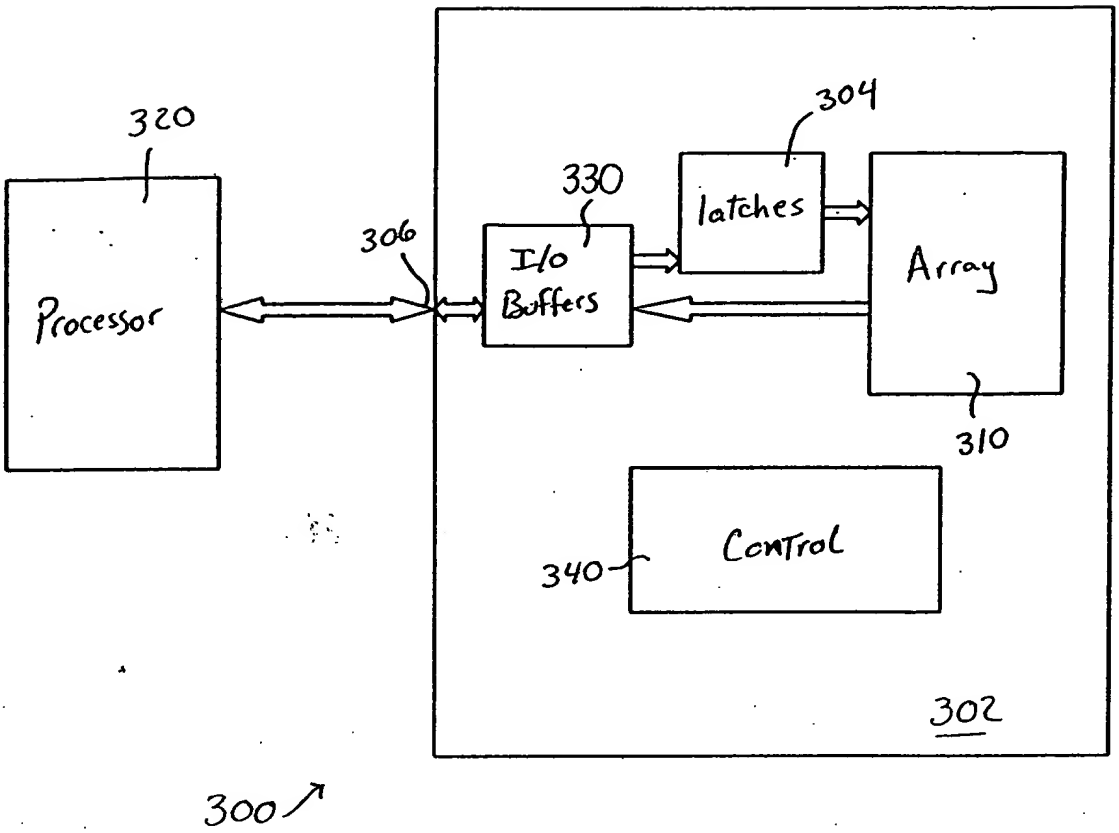

320
Processor
306
330
I/O Buffers
304
latches
Array
310
Control
340
302
300

Fig. 32

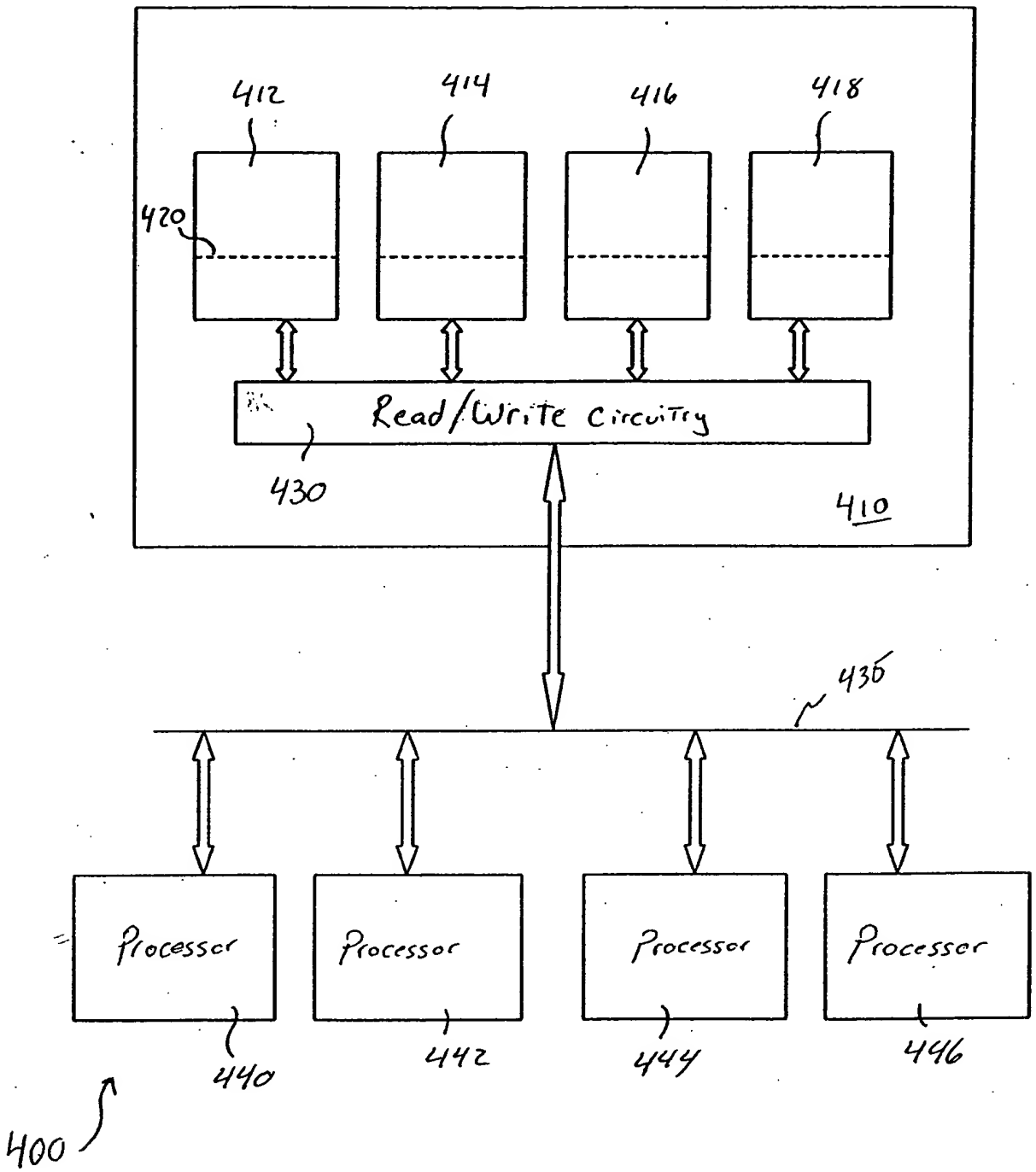

412
414
416
418
420
Read/Write Circuitry
430
410
436
Processor
Processor
Processor
Processor
440
442
444
446
400

Fig. 33

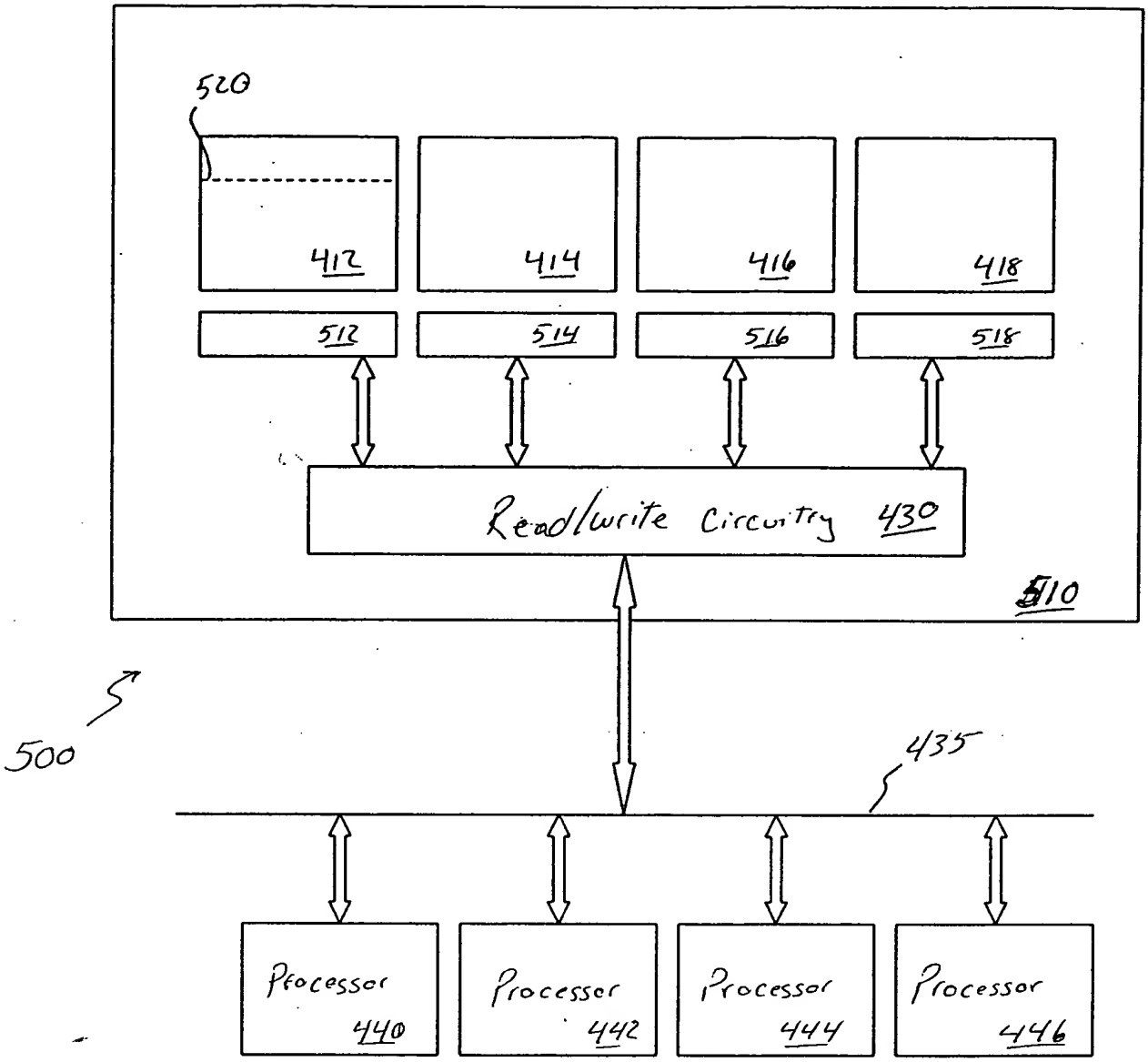

520
412
414
416
418
512
514
516
518
Read/write Circuitry 430
510
500
435
Processor 440
Processor 442
Processor 444
Processor 446

Fig. 34